LDSScanner: Exploratory Analysis of Low-Dimensional Structures in High-Dimensional Datasets

Jiazhi Xia, Fenjin Ye, Wei Chen*, Yusi Wang, Weifeng Chen, Yuxin Ma, and Anthony K.H. Tung

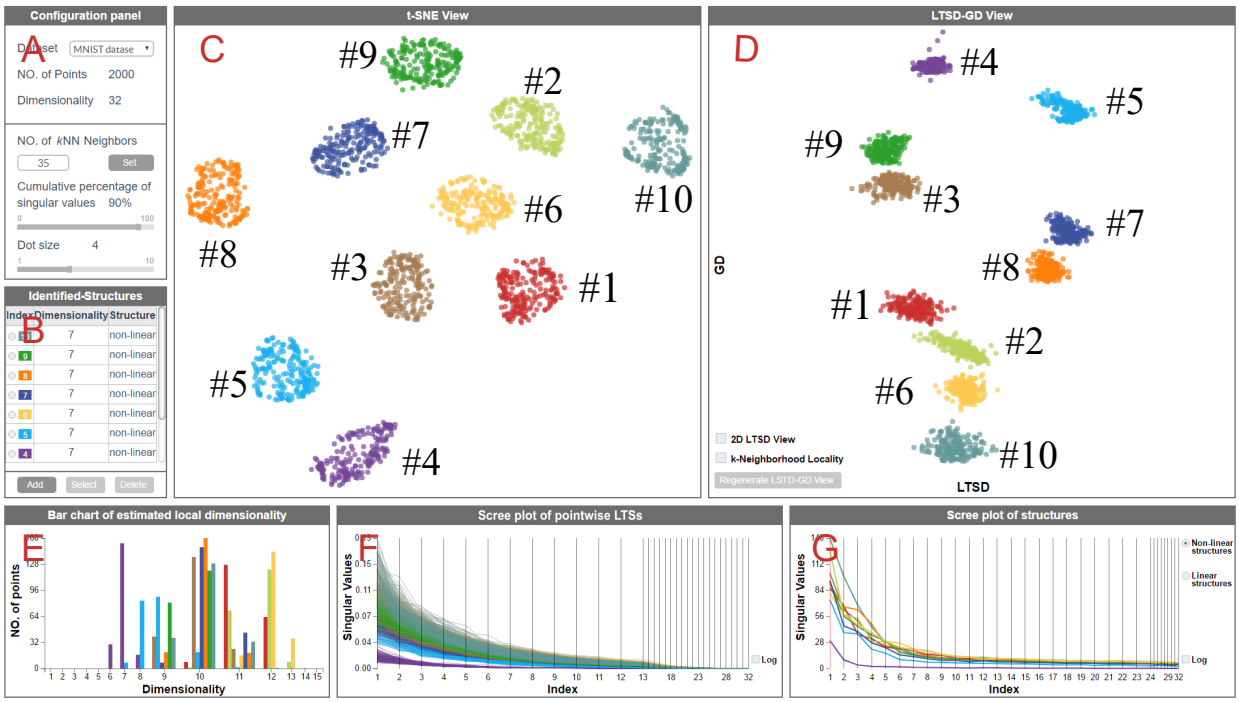

Fig. 1. The exploratory interface of LDSScanner, after the analyst has identified structures. (a)The configuration panel. (b) The identified-structures view. (c) The *t-SNE* view. (d) The LTSD-GD view. (e) Bar chart of estimated local dimensionality. (f) Scree plot of pointwise LTS. (g) Scree plot of structures.

Abstract—Many approaches for analyzing a high-dimensional dataset assume that the dataset contains specific structures, e.g., clusters in linear subspaces or non-linear manifolds. This yields a trial-and-error process to verify the appropriate model and parameters. This paper contributes an exploratory interface that supports visual identification of low-dimensional structures in a high-dimensional dataset, and facilitates the optimized selection of data models and configurations. Our key idea is to abstract a set of global and local feature descriptors from the neighborhood graph-based representation of the latent low-dimensional structure, such as pairwise geodesic distance (GD) among points and pairwise local tangent space divergence (LTSD) among pointwise local tangent spaces (LTS). We propose a new LTSD-GD view, which is constructed by mapping LTSD and GD to the *x* axis and *y* axis using 1D multidimensional scaling, respectively. Unlike traditional dimensionality reduction methods that preserve various kinds of distances among points, the LTSD-GD view presents the distribution of pointwise LTS (*x* axis) and the variation of LTS in structures (the combination of *x* axis and *y* axis). We design and implement a suite of visual tools for navigating and reasoning about intrinsic structures of a high-dimensional dataset. Three case studies verify the effectiveness of our approach.

Index Terms—High-dimensional data, low-dimensional structure, subspace, manifold, visual exploration

1 Introduction

Usually, high-dimensional data is composed of several low-dimensional structures, such as clusters in linear subspaces or non-linear manifolds. A large number of automatic approaches have been proposed for detecting the intrinsic low-dimensional structures. However, specifying appropriate models and parameters relies on correct assumptions about the intrinsic structures, which is a hard task [26]. In this paper, we propose LDSScanner, an exploratory visual analysis approach that provides contextual information required to select an appropriate model, interprets its results, and tunes its configurations.

- *Jiazhi Xia, Fenjin Ye, and Yusi Wang are with Central South University, Email: {xiajiazhi, yefenjin, yswang}@csu.edu.cn;*
- *Wei Chen and Yuxin Ma are with Zhejiang University, Email: chenwei@cad.zju.edu.cn, mayuxin@zju.edu.cn. Wei Chen is corresponding author;*
- *Weifeng Chen is with Zhejiang University of Finance & Economics, Email: cwf818@gmail.com.*
- *Anthony K. H. Tung is with National University of Singapore, Email: atung@comp.nus.edu.sg.*

Manuscript received 31 Mar. 2017; accepted 1 Aug. 2017.
Date of publication 28 Aug. 2017; date of current version 1 Oct. 2017.
For information on obtaining reprints of this article, please send e-mail to: reprints@ieee.org, and reference the Digital Object Identifier below.
Digital Object Identifier no. 10.1109/TVCG.2017.2744098

Dimensionality reduction (DR) is a perplexing problem, as witnessed by a large literature on model construction, parameter configuration and performance optimization [13]. This complexity may even be aggravated when the goal is not to explore the high-dimensional space, but to discover the concrete subspace structure [40]. As shown in Table 1, different groups of approaches are designed to match distinct intrinsic structures. Selecting an appropriate method requires understanding of intrinsic low-dimensional structures of data. For instance, if points lie in a single linear subspace, the widely-used principal components analysis (PCA) [15] works well. Manifold learning algorithms such as ISOMAP [34] are applied in the situation that the points lie in a non-linear manifold. Likewise, when the points lie in multiple subspaces or manifolds, we need to employ subspace clustering algorithms (e.g., Sparse Subspace Clustering [7]) or manifold clustering algorithms (e.g., Sparse Manifold Clustering [8]). This is complicated when there is a mixture of linear structures (i.e., clusters in subspace) and non-linear structures (i.e., manifolds) in a dataset. Furthermore, many subspace clustering and manifold clustering models demand the specification of latent low-dimensional structures, such as the number and intrinsic dimensionality of the clusters.

Table 1. Models for detecting low-dimensional structures in high-dimensional data.

Number of Subspaces / Manifolds	Linear	Non-linear
Single	Linear DR	Manifold Learning
Multiple	Subspace Clustering	Manifold Clustering

We argue that the exploration of the latent low-dimensional structures is a critical step in high-dimensional data analysis, before the appropriate model, as well as accurate parameter configuration, can be applied [42]. To our best knowledge, there are a few available tools that support such an efficient exploration. Well-studied automatic tools, such as DR, clustering models, intrinsic dimensionality estimators, and so on, generally rely on the assumption that intrinsic structures are known. Their output is usually abstract and deterministic [13]. Visualization is naturally a means to augment explanation and exploration of the process and results when the situation is fuzzy or uncertain [37]. However, traditional visualization methods, such as parallel coordinates and scatterplot matrices, are capable of showing correlations among dimensions, but not the latent structures in subspaces. Projections based on DR contain two stages: prior to selecting appropriate models and parameters, intrinsic structures are identified by the analyst. This is actually a chicken-and-egg problem: without knowing intrinsic structures, the choice of models could be too great to choose amongst; or the visualization could be used to observe the results, but cannot augment the model selection decision. In other words, visualization as a **post-processing** tool can reveal possible mismatching between intrinsic structures and models, but it is highly improbable that it could guide the model selection. Alternatively, visualization can be leveraged as an **intra-processing** aid for exploring and analyzing the structures in subspaces [43], but its efficiency is greatly limited by the capability of the analyst and the efficiency of the visualization techniques employed. We argue that before the analysis we really need a **pre-processing** tool that supports exploratory diagnosis of the underlying data for the purposes of specifying appropriate models.

We propose LDSScanner, a visual diagnosis approach that supports identifying intrinsic low-dimensional structures in high-dimensional data, and specifying detailed context and model configurations. Inspired by manifold learning methods, we characterize intrinsic structures and features of high-dimensional data on the basis of a neighborhood graph structure. In particular, we employ pairwise geodesic distances (GD) among points, which are abstracted from the neighborhood graph, as the global feature descriptors of the latent low-dimensional structure. Meanwhile, we compute the local feature descriptors including local tangent space (LTS) and local tangent space divergence (LTSD), which are captured from the neighbors of individual data points in the high-dimensional data. We also propose a novel 2D saliency map that is constructed by plotting the pointwise local tangent space divergences and geodesic distances of all points. This LTSD-GD view facilitates revealing critical patterns of low-dimensional structures, such as the number of clusters, distribution of subspaces, and the variation of local tangent space direction on the manifold. We additionally design and implement a suite of interactive visualization tools to support interactive exploration of the intrinsic structures and features: a *t-SNE* [39] view that indicates the cluster identification, a scree plot [13] of pointwise LTSs, a scree plot of low-dimensional structures, a bar chart of estimated local dimensionality, and an identified-structures view that records the diagnosis of low-dimensional structures.

The main contributions of this paper are twofold:

- A novel LTSD-GD view that supports exploring the latent low-dimensional structures;
- An exploratory visual analysis framework that facilitates studying intrinsic low-dimensional structures in high-dimensional data.

The rest of this paper is organized as follows. Section 2 reviews the related work. Section 3 presents the problem characterization. Section 4 illustrates our approach. In Section 5, we present three case studies. We discuss the limitations and future work in Section 6 and conclude this paper in Section 7.

2 Related Works

In this section, we discuss the literature on visualizing and exploring high-dimensional data.

2.1 Visualizing high-dimensional data

High-dimensional data visualization has attracted much attention [18]. Conventional visualization techniques, including parallel coordinates [14], scatterplot matrices, and radviz [11], are designed to present the data's statistics, such as the distribution of data in a dimension, or the correlation between two dimensions. However, it is highly improbable that those approaches could reveal the latent cluster patterns and underlying features.

Projection based on DR denotes a big family of visualization methods. Usually, points are projected to two- or three-dimensional space while preserving a certain structure. The widely used PCA tries to preserve the variance of data. The classic Multidimensional Scaling (MDS) [6] preserves the Euclidean distance in the high-dimensional space during the projection. Those approaches assume that the points lie in a single low-dimensional linear subspace. If the points lie in a non-linear manifold, several manifold learning approaches have been proposed to detect the latent manifold, such as ISOMAP [34], LLE [24], LE [4], and LTSA [44]. However, projections based on DR can be hard to understand and interpret [31]. First, the points are blindly reduced to two- or three-dimensional space, which may not accurately capture the intrinsic structure due to limited dimensionality [13]. Second, the analyst may lack knowledge of the match between the DR approach and intrinsic structure [13]. The analyst might has a limited understanding of DR models or has a limited knowledge of data at the initial stage.

In a real-world high-dimensional dataset, the points often lie in multiple subspaces or manifolds. Subspace clustering [40] and manifold clustering [8] approaches are designed to detect underlying multiple low-dimensional structures. The common strategy is still to adopt a DR model first and visualize the detected low-dimensional structures subsequently. The VISA [2] system proposed the similarity between clusters in terms of the size of clusters and dimensions in a global view. It also shows the properties of individual clusters in a detailed view. Tatu et al. [33] developed a visualization and navigation interface to explore the large sets of subspaces found by SURFING [3]. The subspaces are organized based on a similarity function that focuses on the topological and dimensional overlap between subspaces. Liu et al. [19] extracted the subspaces with a spectral method and provided a visualization to present the subspaces. A navigation graph and smooth morphing between subspaces were proposed. However, the subspace clustering and manifold clustering models often strongly depend on assumptions of the intrinsic structures, such as the number of clusters [7], the dimensionality of clusters [36], and non-intersection

between structures. It is hard to choose an appropriate model without having knowledge of the intrinsic structure.

Among these visualization methods, *t-SNE* [39] makes a mild assumption about intrinsic structures. In the low-dimensional embedding, it preserves the statistics on the k-Nearest-Neighbors (kNN) graph which is employed for different structures, such as multiple manifolds. Because *t-SNE* preserves the local structure and shows global information [22], we use it as one of the initial indicators of the intrinsic structure in the high-dimensional dataset and optimize its embedding in the visual exploratory process. For computation efficiency, a hierarchical strategy can be adopted [21] in *t-SNE*.

2.2 Exploring high-dimensional data

Exploring high-dimensional data requires pre-processing tools that rely on interactions [42] more than automatic models. Traditional approaches, such as parallel coordinates, scatterplot matrices [12], and various projections [25] are well studied.

Usually, the analyst needs to generate numerous views interactively during the exploration. To ease the manual burden, Voyager [41] proposed automatically generating visualization recommendations. Following this line, there is a large literature on the quality measuring [5, 32] of and ranking [29] visualizations. DimScanner [42] proposed structuring the visualizations to uncover information-aware relations among different views. Sarvghad et al. [27] visualized the dimension coverage to record the exploratory history. To explore the salient subspaces, Yuan et al. [43] proposed a matrix/tree structure to organize the subsets of data and dimensions during the exploratory process.

To enhance the capability of revealing the latent structure, automatic DR models are integrated into the exploration interface. Because the latent structure of data is often unknown, those approaches utilize hybrid models [23] to match the latent structure. DimStiller [13] denoted a work flow for dimensional analysis and reduction combining automatic DR models. Often, only models that are relatively easy to understand can be chosen, such as PCA and MDS. When dealing with data that contains multiple low-dimensional structures, it is highly improbable that these models and workflows could yield an interpretable result. In this paper, we seek to characterize the intrinsic structures and features before constructing the concrete model. The structured features and visualizations disclose latent low-dimensional structures.

3 Problem Characterization

In this section, we describe the representation of low-dimensional structures, illustrate the analysis tasks, and present the feature characterization.

3.1 Representation of low-dimensional structures

We define the salient subspace as the basis for a low-dimensional representation of a cluster [40]. Given a set of points drawn from a linear or affine subspace, the subspace can be fit by Singular Value Decomposing (SVD) or Principal Components Analysis (PCA). Note that the basis can be either axis-aligned or non-axis-aligned.

In more general cases, points lie in non-linear manifolds. We can consider a manifold as a "soft" and "curved" subspace. Each point of a d-dimensional manifold has a small neighborhood that is homeomorphic to the Euclidean space of dimension d. In a manifold, the directions of pointwise local subspaces change gradually. On the other hand, a subspace is a general case of a manifold. For a cluster in a subspace, the directions of pointwise local subspaces are identical to each other.

Inspired by manifold learning approaches, we build a neighborhood graph, e.g., a kNN graph, to represent the clusters that lie in latent manifold or subspace. This representation is based on a locality assumption [16] that for each point there is a small neighborhood in which only the points that belong to the same manifold lie approximately in a low-dimensional affine subspace. This means that even in the full-dimensional space, the local subspace of each point can be captured from the local neighborhoods. Given an unfamiliar dataset, the analyst can always start from a uniform representation.

In this paper, we use a Shared-Nearest-Neighbor (SNN) graph [9], which is a subgraph of a kNN graph to represent the intrinsic structure. In an SNN graph, there is an edge between two points p and q if and only if p and q lie in the kNN neighborhood of each other. An SNN graph tends to generate multiple partitions, which is very suitable for presenting clustering patterns.

3.2 Tasks of low-dimensional structure analysis

We survey the literature, including DR, subspace clustering, manifold learning, and manifold clustering, to conclude the major contextual information required to choose automatic models and tune parameters. We also work closely with an expert in data mining to confirm the analysis tasks.

T1. The number of subspaces/manifolds in the data. For clustering tasks, the first question is how many clusters are in the dataset. This number is often needed as a parameter for clustering algorithms [7].

T2. Is it a linear or non-linear structure? Usually, the analyst does not know if the points lie in a linear subspace or a non-linear manifold. However, linear dimensionality reduction or subspace clustering approaches cannot capture the manifold structure. On the contrary, it is highly improbable that manifold learning and manifold clustering approaches could capture the global linear structure and they may yield undesired results.

T3. The intrinsic dimensionality of subspaces/manifolds. Intrinsic dimensionality is one of the key features of low-dimensional structures and an important parameter of dimensionality reduction models. There are many automatic intrinsic dimensionality estimators that do not offer additional contextual information such as the eigenvalues in PCA, stress values in MDS, the shape of the intrinsic structure, and the variation of local intrinsic dimensionality. A visual exploratory interface can greatly enhance the situational awareness of the explored high-dimensional space.

T4. What are the distributions of subspaces/manifolds? When there are multiple subspaces and manifolds, their organization is critical for selecting models and parameters. How close are the subspaces/manifolds? What are the principal angles between subspaces? Do the subspaces and manifolds intersect each other? Answers to these questions help the analyst reason about the difficulty of identifying low-dimensional structures. This information can also guide the choosing and tuning of automatic models.

T5. Does the locality assumption hold? Our approach is based on the locality assumption [16] that the local subspace of a point can be fit by its neighbors. Under-sampling and noise would breach the locality assumption, leading to results that are hard to understand even if an appropriate model is performed. Exploring the k-neighborhood locality could help the analyst to understand the reason for the failure in visualization.

3.3 Feature characterization in low-dimensional structures

Salient geometry and topology features are characterized based on the SNN representation of low-dimensional structures.

Partitions of SNN graphs. We use SNN graphs to facilitate the separation of points in different structures. Although the components of SNN graphs may not imply an accurate partition, they do provide a good initial estimation to indicate the clustering. If two points are disconnected in an SNN graph, they are regarded as members of two clusters.

Geodesic distance (GD). In manifold learning approaches, pairwise geodesic distances among points are widely used to measure the global intrinsic feature. For instance, the ISOMAP [34] embeds the manifold in a low-dimensional space by preserving the geodesic distances, and reveals the intrinsic features of the manifold. Technically, the geodesic distance is approximated by the shortest distance in the SNN graph.

Local tangent space (LTS). We assume that for each point p there is a small neighborhood that contains only points of the same manifold [8]. The k-nearest neighborhoods spans the local tangent space $\mathbb{S}_p$ of p. We fit $\mathbb{S}_p$ by performing an SVD on the neighbors [44]. Given the

neighborhood matrix X with n rows and d columns, where each row x_i denotes a neighboring point and d represents the dimensionality, we have $X = U\Sigma V^T$, where Σ is a diagonal matrix constructed by the singular values $\{\sigma_1, ..., \sigma_n\}$ listed in descending order. We choose the first d_p singular values while $\frac{\sum_{i=1}^{d_p}\sigma_i}{\sum_{i=1}^{d}\sigma_i} \geq \alpha$, where α is set as 0.9, and the analyst can adjust it manually. The local tangent space $\mathbb{S}_p$ is spanned by the right singular vectors in V corresponding to d_p singular values. Here, d_p is the **estimated local dimensionality**.

Local tangent space divergence (LTSD). Given the local tangent space of the points, we can measure the divergence between two local tangent spaces $\mathbb{S}_p$ and $\mathbb{S}_q$. Here, we introduce three definitions [30].

Definition 1. There are $min(d_p, d_q)$ principal angles between two subspaces $\mathbb{S}_p$ and $\mathbb{S}_q$ of dimensions d_p and d_q. The principal angles are recursively defined. Formally, the ith principal angle is defined as $cos(\theta^{(i)}) = \max_{u\in\mathbb{S}_p^{(i)}} \max_{v\in\mathbb{S}_q^{(i)}} u^T v = u_i^T v_i$, where u and v are normalized orthobases for $\mathbb{S}_p$ and $\mathbb{S}_q$, $\mathbb{S}_p^{(i)} = \mathbb{S}_p - u_1 - ... - u_{i-1}$, $\mathbb{S}_q^{(i)} = \mathbb{S}_q - v_1 - ... - v_{i-1}$, and u_i and v_i are the vectors when the ith maximum value of $u^T v$ is achieved. In practice, we can compute the cosine of the principal angles as the singular values of $B_p^T B_q$, where B_p is a normalized orthobase matrix of $\mathbb{S}_p$ and B_q is a normalized orthobase matrix of $\mathbb{S}_q$.

Definition 2. The normalized affinity between two subspaces is defined as

$$aff(\mathbb{S}_p, \mathbb{S}_q) = \sqrt{\frac{cos^2\theta^{(1)} + ... + cos^2\theta^{(d_p \wedge d_q)}}{d_p \wedge d_q}} \quad (1)$$

The affinity is low when the principal angles are nearly right angles, and vanishes when two subspaces are orthogonal. It is high when the principal angles are small, and is 1 when one subspace lies in the other.

Definition 3. The divergence between two subspaces is defined as

$$div(\mathbb{S}_p, \mathbb{S}_q) = 1 - aff(S_p, S_q) \quad (2)$$

While $aff(\mathbb{S}_p, \mathbb{S}_q)$ denotes the principal angles between two local subspaces, in a subspace cluster, the divergences among the local tangent spaces of point are small. In a manifold, the local tangent spaces of points vary smoothly.

***k*-Neighborhood Locality.** Violations of the locality assumption can be introduced by under-sampling and noise. Here, we define k-neighborhood locality to measure whether point p lies in the local tangent space as $L_p = dis_p / dis_n$, where dis_p denotes the distance from p to the fit local tangent space $\mathbb{S}_p$, dis_n denotes the average distance from p to its neighbors for normalization. As shown in Fig. 2, when p is far from $\mathbb{S}_p$, L_p approximates to 1; when p lies in $\mathbb{S}_p$, L_p equals 0.

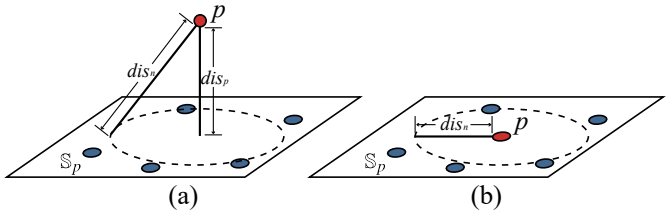

Fig. 2. Estimation of the k-neighborhood locality of a point, which indicates whether it lies in its fit local subspace. (a) The point p does not lie in its local subspace; (b) The point p lies in its local subspace.

4 Design Guidelines

We formulate the design guidelines by interacting closely with a data mining expert. First, a wide range of high-dimensional data visualization and exploration tools are considered. Their capabilities and challenges are summarized. Subsequently, we formulate the design decisions and implement the prototypes during iterative discussions.

G1. The visual design should be easy to understand by analysts with different understandings of data mining models. There are two considerations for the visual design. First, the output of automatic approaches are too abstractive to be understood, demanding expressive visual encodings. Second, structuring high-dimensional spaces should employ the popular visual forms that are easily recognizable for domain experts.

G2. The system should provide contextual information of the low-dimensional structure. Through the pilot interview with domain experts, we realize that it is important to provide contextual information to the analyst, rather than a conclusion. It helps the analyst to understand the dataset and the underlying mathematics of feature characterization, and at the same time, evaluate the reliability and uncertainty of the characterized feature.

G3. The exploratory process should be traced and recorded. High-dimensional data exploration is often a highly free and iterative process. The analyst may become lost in the exploration iterations. As such, a status reporting option can be used to profile, share, and reuse the analysis process.

5 Visualization Design

In this section, we overview our interface and explain how the analysis tasks are supported by each view in a typical workflow. Subsequently, we describe the detailed design of each view.

5.1 Overview

Our visual exploration interface consists of multiple views (Fig. 1). We introduce these views according to the order of a typical workflow.

First, after the dataset and system parameters are specified in the configuration panel (Fig. 1(a)), the *t-SNE* view (Fig. 1(c)) presents a 2D non-linear embedding of data points and usually performs well in visual clustering. The number of subspaces/manifolds is visually presented (T1). The *t-SNE* view supports *lasso*-based selection which is linked with other views.

Next, the analyst can explore the LTSD-GD view (Fig. 1(d)), which is designed to support analysis tasks T2, T4, and T5. In the LTSD-GD view, the analyst can explore a subset of points. After selecting a subset of points in the *t-SNE* view or LTSD-GD view and clicking the "Regenerate LTSD-GD view" button, a new layout of the LTSD-GD view for selected points is generated. If a low-dimensional structure is identified, the analyst can add it into the list of identified-structures (Fig. 1(b)).

The intrinsic dimensionality is another important concern for the analyst (T3). To provide the contextual information, we present three views. The analyst starts by analyzing the local intrinsic dimensionality of LTS. The scree plot of pointwise LTS (Fig. 1(f)) provides ordered eigenvalues of each LTS. The trend of the eigenvalues is helpful to understand local intrinsic dimensionality. Then, the histogram of estimated local dimensionality (Fig. 1(e)) shows the statistics of the local dimensionality. This view is connected to other views with bin-based selection. Having a good understanding of the local dimensionality, the analyst can study the intrinsic dimensionality of structures in the scree plot of structures (Fig. 1(g)). In this view, eigenvalues from SVD decompositions of the structures are presented.

5.2 The LTSD-GD view

In manifold learning and manifold clustering approaches, it is quite common but critical to characterize the local tangent space of points. To our best knowledge, there is no such tool to visualize the distribution of local tangent spaces. We aim to design a tool to address this problem by visually coding the local tangent space information.

In the first round design of the 2D LTSD view, we project the points into a 2D space using 2D MDS with the LTSD information. Points lying in the identical subspace have similar local tangent spaces, and are gathered in the projection. In the case that points lie in a manifold, the local tangent spaces of the points vary smoothly. Points are scattered in the projection. It can be seen as a visual indication that the manifold is non-linear. However, the diversity of pointwise LTS may be caused

by noise rather than variation in the manifolds. This design cannot distinguish these two cases.

Inspired by ISOMAP [34], we seek to unfold the manifold and visualize the variation of LTS along a certain intrinsic 1D dimension or 2D plane. We decide to unfold the manifold in the y axis. Meanwhile, we encode the LTSD information in the x axis. We call this design the LTSD-GD view. Similar to the 2D LTSD view, the x axis presents the distribution of the local tangent spaces of points, e.g., linear or non-linear structures. Along the y axis, each cluster is unfolded along the most informative "direction". Meanwhile, clustering information is visually revealed because points in different clusters are disconnected from each other. The most interesting part is the combination of the x axis and the y axis. Given a manifold, neighboring points are close to each other on the y axis and the variations of the LTSs of them are shown in the x axis. Therefore, the view shows an oblique line if the LTSs of the points vary uniformly, or a curve if the LTSs of the points vary non-uniformly. If the diversity of the LTS is caused by noise, the LTSD-GD view contains noise.

1D MDS may lead to information loss. We keep the two designs and take the LTSD-GD view as the primary view. In most cases of our experiments, the LTSD-GD view is adequate to distinguish different clusters and subspaces.

5.2.1 Construction of the view

The x axis shows the distribution of the pointwise LTSs. First, we compute the LTS of each point by performing SVD on its neighbors (see Section 3.3). Second, for each pair of points, we measure the LTSD between the pointwise LTS of them (Equation 2). Third, while the pairwise LTSD measures a certain distance between LTSs of two points, we use a 1D MDS algorithm to map points to the x axis with respect to the pairwise LTSDs among all points.

The y axis encodes the geodesic distances among points. Intuitively, we can employ ISOMAP to project points to the y axis. However, ISOMAP fails to capture the intrinsic structure when there are multiple manifolds in the dataset. To address this problem, we propose employing a hierarchical 1D MDS to map the manifolds into a 1D space and combine the mappings together.

The algorithm is shown in Fig. 3. Firstly, we partition the points according to the underlying SNN graph (Fig. 3(a, b)). The partition result implies the clustering information. Secondly, we treat each partition as a point and project it into 1D space by MDS (Fig. 3(c)). Thirdly, in each partition, the points are projected to a 1D space by using the 1D MDS algorithm subject to the geodesic distances (Fig. 3(d)). Fourthly, we align the orientation of partitions in the 1D space according to the inter-partition distances (Fig. 3(e)). Finally, we scale the length of the 1D projection to 1.

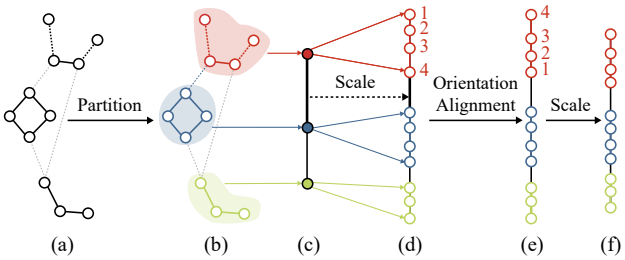

Fig. 3. Illustration of hierarchical 1D MDS.

***k*-Neighborhood Locality.** We encode the k-neighborhood locality with the opacity of points in the LTSD-GD view as an option. As described in Section 3.3, the pointwise locality loc ranges from 0 to 1. When the locality model is enabled, the opacity of the points is modulated as $opac(0.9loc + 0.1)$ where $opac$ denotes the original opacity.

5.2.2 Visualizing low-D structures with the LTSD-GD view

The angle between subspaces. We verify the capability of the LTSD-GD view with a set of synthetic datasets. They contain two 2D subspaces embedded in a 3D space. The angles between the subspaces are 0, $\pi/6$, $\pi/3$, and $\pi/2$, respectively (see Fig. 4(a)–(d)). Fig. 4(e)–(h) show LTSD-GD views of these datasets. Two lines in the view indicate that there are two clusters (T1). The vertical line pattern indicates that the local tangent spaces are identical and the points lie in a linear subspace (T2). The distance between two clusters on the LTSD axis encodes the angle between the corresponding subspaces (T4). The closer the angle is to $\pi/2$, the farther the clusters are from each other on the LTSD axis. In Fig. 4(e), two clusters are located near in LTSD axis. This implies that they lie in the same linear subspace.

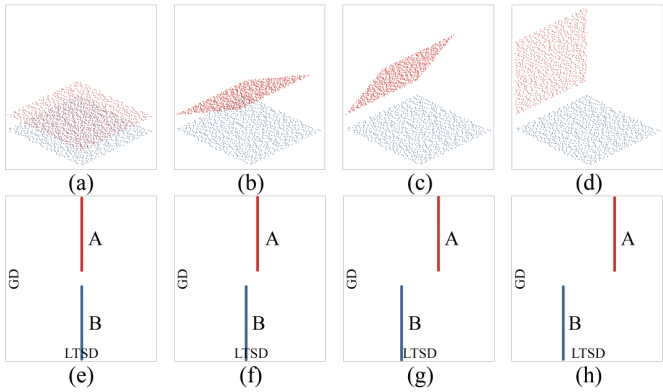

Fig. 4. The angle between the subspaces. (a)–(d) The angles between the two 2D subspaces are 0, $\pi/6$, $\pi/3$, and $\pi/2$, respectively. (e)–(h) The LTSD-GD views of the dataset in (a)–(d), respectively.

Intersected subspaces and manifolds We examine three synthetic datasets that contain intersected subspaces and manifolds. Each dataset contains two low-dimensional structures. Because two structures intersect, they are identified as one cluster in the LTSD-GD view. To study their patterns, we color them in red and blue, respectively.

Fig. 5(d) shows the LTSD-GD view of two intersected subspaces (see Fig. 5(a)). The LTSD-GD view shows two main subspaces represented by two vertical lines A and B (T1 and T2). Breaks in the middle of lines suggest that there are sudden changes. By observing the points at the middle of two lines, we make a hypothesis that these points are in the intersecting section. This view supports preliminary verification of hypothesis (T4): 1) the distribution in the GD direction suggests that these points are connected to two main low-dimensional structures; 2) the distribution in the LTSD direction indicates that the local tangent spaces of these points is located between two main subspaces.

The second dataset contains an S-shaped manifold that intersects with a subspace (see Fig. 5(b)). Fig. 5(e) shows the LTSD-GD projection. The curved shape A in the projection shows the variation of the local tangent spaces in the manifold (T2 and T4). The vertical line B represents a subspace. Similar to the case of intersected subspaces, the intersecting sections are contained in different local subspaces. Similarly, the third dataset contains two manifolds (see Fig. 5(c)). Their patterns are shown in Fig. 5(f).

5.3 Descriptive views

5.3.1 The *t-SNE* view

We choose to integrate the *t-SNE* view to provide an initial indication of clustering information (T1). *t-SNE* does not require knowledge of the specific structure in advance but only makes a mild assumption that the nearest neighbors of a point lie in the same manifold. Therefore, it fits well when the intrinsic structure is unknown.

When a low-dimensional structure is identified, the underlying kNN graph is updated by removing the edges connecting the identified points with other points. We update the *t-SNE* view to reflect the change and enhance the clustering pattern.

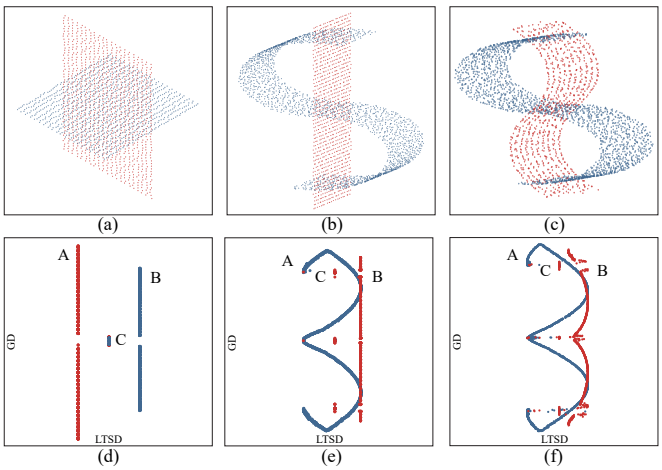

Fig. 5. The LTSD-GD views of intersected subspaces and manifolds. In each view, two structures are colored by blue and red, respectively. (a) Two intersected 2D subspaces are embedded in a 3D space. (b) A 2D manifold intersects with a 2D subspace in 3D space. (c) Two 2D manifolds intersect with each other. (d)–(f) The LTSD-GD view of (a)–(c).

5.3.2 The scree plot of pointwise LTSs

We present the scree plot of pointwise local tangent spaces of individual points to disclose the intrinsic dimensionality. Scree plot is widely used to provide the contextual information of PCA [13] by depicting the numbers and values of dimension factors.

Here, we present the scree plot in the parallel coordinates form. A polyline presents the singular values $\{\sigma_1, ..., \sigma_n\}$ from the SVD decomposition for the local tangent space (see Section 3.3). The singular values are indexed in descending order. We can estimate the dimensionality of the corresponding LTS by studying where the gradient of the line changes suddenly.

To alleviate the scalability problem of parallel coordinates when the dimensionality increases, we shorten the intervals between the axes with small singular values. Specifically, given the scree plot of a group of local tangent spaces, we set the intervals as follows.

Step 1. Count the largest singular value in each axis as $\{\sigma_1^{max}, ..., \sigma_n^{max}\}$, where n is the number of points.

Step 2. Count the number of significant axes n_s as

$$\frac{\sum_{i=1}^{n_s} \sigma_i^{max}}{\sum_{i=1}^{n} \sigma_i^{max}} \geq \alpha, \tag{3}$$

where the value of α is initialized as 0.9, and is adjustable.

Step 3. Set the intervals associated with the first n_s axes as three times the rest.

Step 4. Cut off the axes for which σ^{max} is zero.

5.3.3 The scree plot of low-dimensional structures

We also provide a scree plot of low-dimensional structures to support the estimation of their intrinsic dimensionality. Here, a polyline presents the singular values from the decomposition of a structure. To enhance the dimensionality scalability, the intervals between axes are also dynamically adjusted. Two options are offered to construct the scree plot. The term of linear structures denotes the singular values computed by SVD to the matrix of points in the structure. The term of non-linear structures refers to the singular values obtained by MDS with regard to pairwise geodesic distances among points. Among the variants of MDS, we choose the one addressed by matrix decomposition, because it is faster and more stable than the iterative optimization approach. The following is the algorithm for computing the singular values.

Step 1. Build the kNN graph in the set points in the structure, P.

Step 2. Compute the geodesic distance matrix G, which is $n \times n$, where g_{ij} is the geodesic distance between points p_i and p_j in the kNN graph, and n is the number of points.

Step 3. Compute a $n \times n$ matrix B, where

$$b_{ij} = -\frac{1}{2}(g_{ij}^2 - \frac{1}{n}\sum_{j=1}^{n} g_{ij}^2 - \frac{1}{n}\sum_{i=1}^{n} g_{ij}^2 + \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n} g_{ij}^2) \tag{4}$$

Step 4 Decompose B using SVD: $B = U\Lambda U^T$, where Λ is a diagonal matrix constructed by the singular values $\{\lambda_1, ..., \lambda_n\}$, which are listed in descending order.

This algorithm has an intuitive explanation similar to KPCA [28]. We can project points into a high-dimensional space in which the Euclidean distances between points are equal to the geodesic distances in the kNN graph. If the representation of points in this high-dimensional space is known, we denote the points as matrix X, where each row x represents a point. We can perform dimensionality reduction by decomposing X. However, the representation of X is unknown because it is hard to recover the projection. Alternatively, we can build the matrix $B = XX^T$ from the distance matrix D and decompose B to conduct the dimensionality reduction.

The analyst can study the dimensionality of structures by the gradient of lines, and infer whether a structure is linear or non-linear by comparing two modes. If a structure is non-linear, the non-linear mode indicates lower intrinsic dimensionality. Otherwise, two modes can lead to a similar dimensionality.

5.3.4 Bar chart of estimated local dimensionality

A bar chart is employed to visualize the distribution of estimated local dimensionality, which is estimated as mentioned in Section 3.3.

5.3.5 The identified-structures view

The goal of this view is to trace the exploration and provide a diagnosis report after the exploration. It records the profiles of low-dimensional structures as a list. Items in the list represent identified low-dimensional structures. Each item contains three fields: the structure ID, the estimated dimensionality, and whether it is a linear subspace or a non-linear manifold. In this view, the analyst can identify, fill, select, and delete items of low-dimensional structures.

5.3.6 The configuration panel

This panel shows the dataset information including the raw dimensionality and number of points. Two global parameters k and α can be manually adjusted. k refers to the size of neighborhood in the kNN graph and SNN graph. When k is modified, the underlying neighborhood graphs and intrinsic features are updated. α refers to the threshold of intrinsic dimensionality estimation. When it is modified, the local tangent spaces are updated. Subsequently, the LTSD-GD view and bar chart of estimated local dimensionality are updated.

6 Case Studies

In this section, we describe how our approach facilitates the diagnosis of high-dimensional datasets and identification of latent low-dimensional structures, through case studies of one synthetic dataset and two real-world datasets.

6.1 The synthetic 12-D dataset

We conduct a case study to explore a 12-D synthetic dataset used in [10, 33]. There are 750 points distributed in six Gaussian clusters.

Our exploration begins with the *t-SNE* view and the LTSD-GD view. The *t-SNE* view shows the pattern of four clusters (Fig. 6(a)). In the LTSD-GD view, it indicates six clusters (Fig. 6(b)) (T1). Through dynamic querying between the two views, we confirm that the left three clusters, *A*, *B*, and *C*, in the *t-SNE* view correspond to three clusters in the LTSD-GD view, respectively. In the LTSD-GD view, clusters *F* and *G* intersect with cluster *E*. In contrast, they are mixed with each other in the *t-SNE* view as *D*. Subsequently, we label the six clusters as low-dimensional structures and update the *t-SNE* view (Fig. 6(c)) and the LTSD-GD view (Fig. 6(d)).

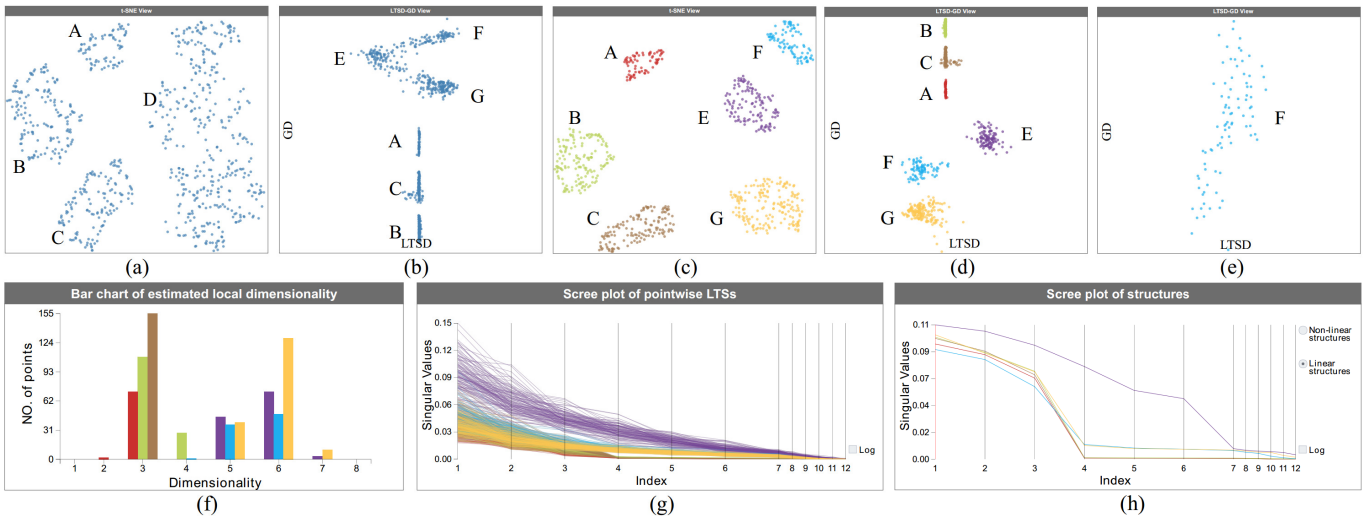

Fig. 6. The synthetic 12-D dataset. (a) The initial *t-SNE* view. (b) The initial LTSD-GD view. (c)–(h) The *t-SNE* view, LTSD-GD view, LTSD-GD view of structure F , bar chart of estimated local dimensionality, scree plot of pointwise LTS, and scree plot of structures, after labeling the low-dimensional structures.

In the LTSD-GD view, we can see that clusters A, B, and C are in the same linear subspace because the points are concentrated on the x axis (T2). In the 2D LTSD view, three clusters are also mapped together, which verifies our inference. F and G are approximately in the same subspace (T4). The intra-distances of them are larger than those of A, B, and C. To verify whether they are linear or non-linear (T2), we choose each structure and generate a new LTSD-GD view. Fig. 6(e) shows the LTSD-GD view of structure F. It shows an approximately vertical pattern with much noise. This might be caused by the intersection among them, which is shown in Fig. 6(c). In general, we infer that all six clusters are in three linear subspaces. A, B, and C share a subspace and F and G lie in another subspace, while E lies in a distinct subspace.

To estimate the intrinsic dimensionality of the six clusters, we study the bar chart and two scree plots (T3). The bar chart of estimated local dimensionality (Fig. 6(f)) shows that the local dimensionalities of A, B, and C are three. Most points in clusters E, F, and G have local dimensionalities of five or six. The scree plots provide the contextual information to verify the intrinsic dimensionality (G2). In the scree plot of pointwise LTS (Fig. 6(g)), the polylines of points in E, F, and G change around the sixth axis. The scree plot of structures (Fig. 6(h)) provides the contextual information in the aspect of structure. The polyline corresponding to E has a sudden change in the sixth dimension. This indicates that the intrinsic dimensionality of E is six. The polylines of F and G have a tuning in the fourth dimension and a small tuning in the sixth dimension. Considering the information in the bar chart and the scree plot of pointwise LTS, we tend to infer that the dimensionality of F and G is six. Here, we select the "Linear structures" option because linear and non-linear modes present similar patterns and indicate the same intrinsic dimensionality. This verifies that all six structures are linear.

Finally, we record our inference in the identified-structures view and save it as a diagnosis report.

6.2 The Hopkins 155 dataset

In the second case study, we consider the *cars10* video sequence in the Hopkins 155 dataset [35], which contains three motions that lie in three low-dimensional subspaces, respectively. The dataset contains 297 points of 62 dimensions. Each data point refers to a feature point that is tracked through the frames of the video. It encodes the x and y values of the point in 31 frames into a 62-dimensional vector. The points sharing a rigid motion lie in an affine subspace. Therefore, the identification of the subspaces favors addressing the motion segmentation problem.

The LTSD-GD view (Fig. 7(a)) shows that there are three clusters (T1), which are concentrated on the x axis. It indicates that the clusters are almost in the same subspace (T2, T4). The scree plot of pointwise LTS (Fig. 7(b)) and the scree plot of structures (Fig. 7(c)) show that the singular values are close to zero after the second axis. It indicates that the intrinsic dimensionality of the data is two (T3).

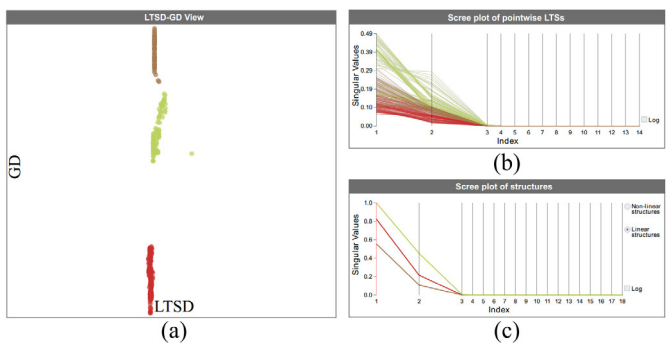

Fig. 7. The Hopkins 155 dataset. (a) The LTSD-GD view of the dataset shows that there are three clusters in the subspaces. (b) The scree plot of pointwise LTS shows that the local dimensionality of the points is two. (c) The scree plot of structures shows the intrinsic dimensionality of the three structures is two.

6.3 The MNIST dataset

We conduct a case study on the MNIST dataset [17] that comprises images of digits $0-9$. The dataset has 60,000 points of 784 dimensions. For each digit, we sample 200 points. The dimensionality is reduced to 32 using PCA. The structures are identified in the initial layout of the *t-SNE* view (see the fifth row of Fig. 9).

The identified-structures view shows that there are ten low-dimensional structures (Fig. 1(b)). The LTSD-GD view (Fig. 1(d)) indicates the variation of the LTSs in each structure. We would like to investigate whether the structures are linear or non-linear (T4). We generate new LTSD-GD views, each of which contains only one structure. For instance, the LTSD-GD view of structure #1 (Fig. 8(a)) shows an oblique line pattern. On the y axis, the structure is unfolded in the first intrinsic dimension. On the x axis, the directions of pointwise LTSs vary smoothly. Therefore, we have a high confidence that this structure is in a manifold. Similarly, the LTSD-GD view of the other nine structures indicates that the corresponding structures are also in manifolds (Fig. 8(b – j)). Considering that the initial LTSD-GD view

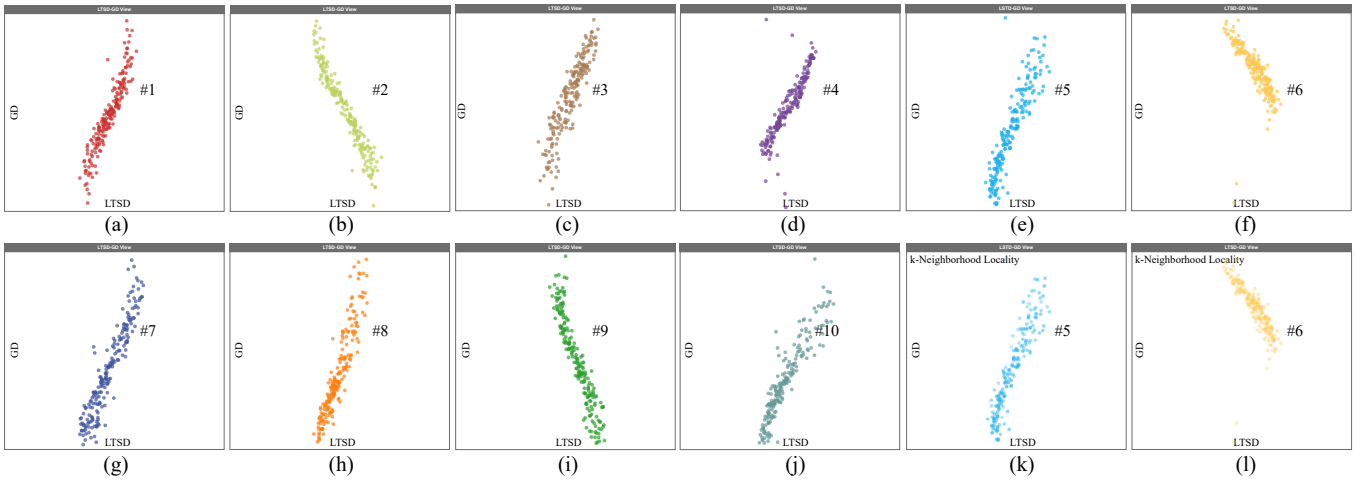

Fig. 8. The MINST dataset with identified structures. (a)–(j) The LTSD-GD views of structures #1 – #10, respectively; (k)(l) The LTSD-GD view of structures #5 and #6 in Locality mode, respectively.

does not present a clear clustering pattern (the bottom left view in Fig. 9), we would like to study how the locality assumption holds. By enabling the option "Locality as transparency" in the LTSD-GD view, we can observe the k-neighborhood locality of points (T5). The transparency of points (Fig. 8(k, l)) shows that the assumption that the kNN of a point lies in the same manifold probably does not hold. Finally, we update the information in the identified-structures view and generate a diagnosis report.

7 Discussion

7.1 The visual design and analysis tasks

In this section, we discuss how the five analysis tasks (Section 3.2) are supported by our visual design. The number of structures (T1) can be detected in the *t-SNE* view and the LTSD-GD view. In most of our case studies, the LTSD-GD view presents clearer cluster patterns than the *t-SNE* view. For the MNIST dataset, the *t-SNE* view achieves better performance (the fifth row in Fig. 9). In practice, these two views are associated with each other to support T1.

For T2, major structural information is present in the LTSD-GD view. The scree plot of the structures also provides supporting evidence for T2 (Section 5.3.3). However, it requires certain domain knowledge. In the future, we would like to study how to present this domain knowledge to novice analysts.

For T3, the analysis of dimensionality is supported by the bar chart of pointwise LTS, the scree plot of pointwise LTS, and the scree plot of structures. The first two views focus on the analysis of local LTS. The last view supports the study of the entire structure. This design follows the strategy of "fit locally, think globally", which is widely used in manifold learning.

For T4, the LTSD-GD view is designed to show the intrinsic distance (GD) on the y axis and the LTSD on the x axis. As a result, it is able to show the angle between subspaces and the intersection among structures (Section 5.2.2).

For T5, how the locality assumption holds is measured by k-neighborhood locality and encoded as opacity in LTSD-GD view.

7.2 Comparisons with visualization techniques

Axis-aligned high-dimensional visualization approaches, such as parallel coordinates and scatterplot matrices, work well in revealing the statistics of high-dimensional data, such as the distribution in a dimension and the correlation between two dimensions. However, they are not designed for clustering and lack clustering-oriented features. Some exploratory approaches, like DimScanner [42] and Voyager [41], engage in organizing and recommending reformulated data organizations to facilitate multi-faceted exploration. Whereas these solutions strive to depict the relation and distribution in terms of data points, they are targeted at the statistics. In contrast, our design is task-oriented. Our approach presents the features concerning the subspaces to support diagnosing the intrinsic structures of high-dimensional data. Thus, our approach makes it amenable for not only previewing the underlying data, but also inspecting important factors in subsequent data analysis tasks.

7.3 Comparison with dimensionality reduction models

There are several differences between LDSScanner and automatic dimensionality reduction models. First, the automatic models are designed to find a low-dimensional embedding that preserves certain intrinsic features, while LDSScanner aims to diagnose the latent low-dimensional structure. Second, automatic models often require knowledge of the intrinsic structures in advance to choose a model and parameters appropriately. LDSScanner proposes exploration of the latent low-dimensional structure as an inspection means. Fig. 9 shows the comparison between our LTSD-GD view and typical automatic models, including PCA (classical MDS is identical to PCA when employing the Euclidean distance), *t-SNE*, ISOMAP, and LLE, with six datasets. The implementations of these methods are based on scikit-learn [20]. The result confirms that ours compares favorably with these counterparts.

7.4 Limitations

Computational complexity. In our implementation, the most time-consuming parts include the kNN graph construction and SVD. A brute-force computation of the exact kNN takes $O(dn^2)$ time complexity, where d is the dimensionality of data and n is the number of points. In addition, the time complexity of SVD is $O(k^3)$, where k denotes the size of kNN, which is proportional to the dimensionality d.

When the dimensionality increases, e.g., more than 100, it is a challenge to achieve interactive performance. In our case studies, the dataset with the highest dimensionality is 64. Interactive performance is achieved with a set of strategies: (1) We choose the vantage-point tree [38] whose time complexity is approximately $O(nlogn)$; (2) We perform dimension reduction in the preprocessing stage. Usually, intrinsic structures can be retained with relatively low dimensionality; (3) We choose a relatively small k. Considering that a small k may struggle to capture the latent structures due to noise and under-sampling, we set $0.5d \leq k \leq 1.3d$ to keep the balance between computational performance and structuring accuracy; (4) Key features are pre-computed, stored in a database, and updated on the fly.

Scalability. Dimensionality can also affect the visual scalability in the scree plots and the bar chart. Fortunately, the latent structures usually have low intrinsic dimensionality. We can hide the bars with

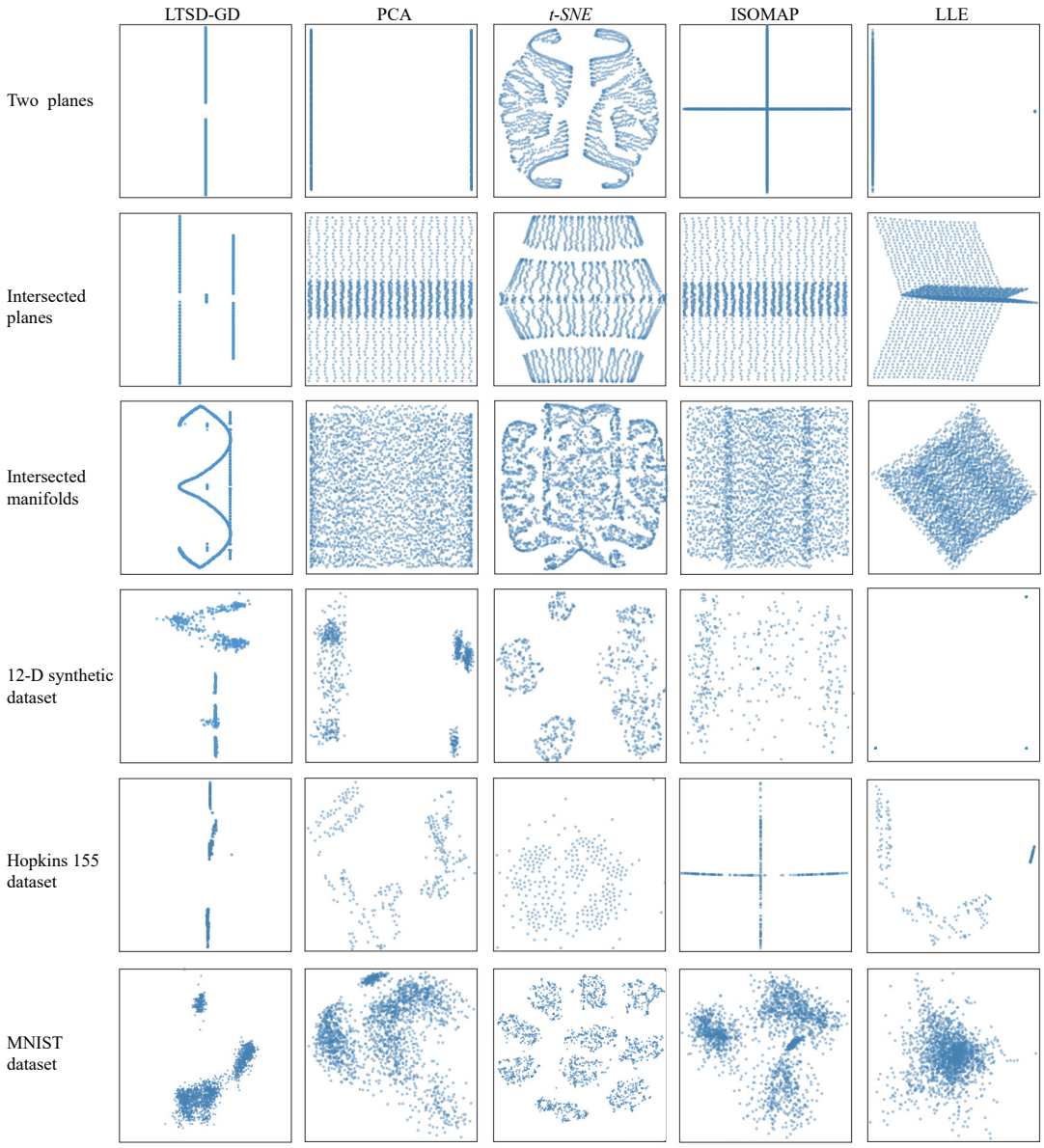

Fig. 9. Comparisons of the proposed LTSD-GD view, PCA, *t-SNE*, ISOMAP and LLE with six datasets that separately contain two planes, intersected planes, intersected manifolds, a 12-D synthetic dataset, the Hopkins 155 dataset, and the MNIST dataset.

zero values in the bar chart. In the scree plot view, the dimensions are ordered by the singular values in descending order. We hide the axes whose maximum singular values are zero.

Sparsity of Data. *k*NN and the locality assumption are widely accepted in manifold learning. However, with complicated distribution of structures, the locality assumption holds only when the data points are densely sampled. Nevertheless, this assumption is not true in many datasets. Our representation of the latent low-dimensional structure is derived from *k*NN and also suffers from the sparsity problem. One possible solution is to perform a global dimension reduction before the subspace analysis. Advanced methods, such as neighborhood range selection [16] and different distance metrics [1], can also be used.

8 Conclusion

In this paper, we propose LDSScanner, a visual analytics approach for previewing and inspecting the low-dimensional structures in high-dimensional space. It empowers the analyst with not only qualitative visual evidences, but also quantitative measurements for hidden low-dimensional structures, for the purposes of effective model selection, parameter modulation, and result justification in analyzing high-dimensional data. We collaborate with a data mining expert to extract the analytical tasks and the design guidelines for system development. Case studies and comparisons demonstrate the effectiveness and efficiency of our approach.

In the future, we would like to verify our approach in a real-world scenario and integrate it with extensive analysis tasks such as deep learning. We also expect to accelerate the implementation by means of parallel computing to support interactive visual analysis of large-scale data.

Acknowledgements

This research is partially supported by National Science Foundation of China (61309009, 61422211), National 973 Program of China (2015CB352503), Major Program of National Natural Science Foundation of China (61232012), and Open Project Program of the State Key Lab of CAD&CG (A1710).

REFERENCES

[1] C. C. Aggarwal, A. Hinneburg, and D. A. Keim. On the surprising behavior of distance metrics in high dimensional spaces. ICDT '01, pages 420–434, 2001.

[2] I. Assent, R. Krieger, E. Müller, and T. Seidl. Visa: Visual subspace clustering analysis. *SIGKDD Explor. Newsl.*, 9(2):5–12, Dec. 2007.

[3] C. Baumgartner, C. Plant, K. Railing, H. P. Kriegel, and P. Kroger. Subspace selection for clustering high-dimensional data. In *ICDM '04*, pages 11–18, Nov 2004.

[4] M. Belkin and P. Niyogi. Laplacian eigenmaps and spectral techniques for embedding and clustering. NIPS, pages 585–591, 2001.

[5] E. Bertini, A. Tatu, and D. Keim. Quality metrics in high-dimensional data visualization: An overview and systematization. *IEEE Transactions on Visualization and Computer Graphics*, 17(12):2203–2212, 2011.

[6] I. BORG and P.J.GROENEN. Modern multidimensional scalling : Theory and applications, 2005.

[7] E. Elhamifar and R. Vidal. Sparse subspace clustering. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 2790–2797, 2009.

[8] E. Elhamifar and R. Vidal. Sparse manifold clustering and embedding. NIPS'11, pages 55–63, 2011.

[9] L. Ertöz, M. Steinbach, and V. Kumar. Finding clusters of different sizes, shapes, and densities in noisy, high dimensional data. In *ICDM*, pages 47–58, 2003.

[10] B. J. Ferdosi, H.Buddelmeijer, S. Trager, M. H. F. Wilkinson, and J. B. T. M. Roerdink. Finding and visualizing relevant subspaces for clustering high-dimensional astronomical data using connected morphological operators. In *IEEE VAST*, pages 35–42, 2010.

[11] P. Hoffman, G. Grinstein, K. Marx, I. Grosse, and E. Stanley. Dna visual and analytic data mining. In *IEEE Vis.*, pages 437–441, 1997.

[12] J. F. Im, M. J. McGuffin, and R. Leung. Gplom: The generalized plot matrix for visualizing multidimensional multivariate data. *IEEE Transactions on Visualization and Computer Graphics*, 19(12):2606–2614, Dec 2013.

[13] S. Ingram, T. Munzner, V. Irvine, M. Tory, S. Bergner, and T. Möller. DimStiller: Workflows for dimensional analysis and reduction. In *IEEE Symposium on Visual Analytics Science and Technology*, pages 3–10, 2010.

[14] A. Inselberg and B. Dimsdale. Parallel coordinates: A tool for visualizing multi-dimensional geometry. In *IEEE Vis.*, pages 361–378, 1990.

[15] I. T. Jolliffe. *Principal Component Analysis*. Springer, 2002.

[16] H.-P. Kriegel, P. Kröger, E. Schubert, and A. Zimek. A general framework for increasing the robustness of pca-based correlation clustering algorithms. In *Proc. SSDBM*, pages 418–435, 2008.

[17] Y. Lcun, L. Bottou, Y. Bengio, and P. Haffner. Gradient-based learning applied to document recognition. *Proceedings of the IEEE*, 86(11):2278–2324, 1998.

[18] S. Liu, D. Maljovec, B. Wang, P.-T. Bremer, and V. Pascucci. Visualizing high-dimensional data: Advances in the past decade. In *Eurographics Conference on Visualization (EuroVis) - STARs*, 2015.

[19] S. Liu, B. Wang, J. J. Thiagarajan, P.-T. Bremer, and V. Pascucci. Visual exploration of high-dimensional data through subspace analysis and dynamic projections. *Comput. Graph. Forum*, 34(3):271–280, 2015.

[20] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and E. Duchesnay. Scikit-learn: Machine learning in Python. *Journal of Machine Learning Research*, 12:2825–2830, November 2011.

[21] N. Pezzotti, T. Höllt, B. Lelieveldt, E. Eisemann, and A. Vilanova. Hierarchical stochastic neighbor embedding. *Computer Graphics Forum*, 35(3):21–30, 2016.

[22] N. Pezzotti, B. Lelieveldt, L. van der Maaten, T. Hollt, E. Eisemann, and A. Vilanova. Approximated and user steerable tsne for progressive visual analytics. *IEEE Transactions on Visualization and Computer Graphics*, PP(99):1–1, 2016.

[23] G. Ross and M. Chalmers. A visual workspace for hybrid multidimensional scaling algorithms. In *IEEE Symposium on Information Visualization 2003*, pages 91–96, 2003.

[24] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290(5500):2323–2326, 2000.

[25] M. Rubio-Snchez, L. Raya, F. Dłaz, and A. Sanchez. A comparative study between radviz and star coordinates. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):619–628, 2016.

[26] D. Sacha, L. Zhang, M. Sedlmair, J. A. Lee, J. Peltonen, D. Weiskopf, S. C. North, and D. A. Keim. Visual interaction with dimensionality reduction: A structured literature analysis. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):241–250, 2017.

[27] A. Sarvghad, M. Tory, and N. Mahyar. Visualizing dimension coverage to support exploratory analysis. *IEEE Transactions on Visualization and Computer Graphics*, 23(1):21–30, 2017.

[28] B. Schölkopf, A. Smola, and K.-R. Müller. Nonlinear component analysis as a kernel eigenvalue problem. *Neural Comput.*, 10(5):1299–1319, 1998.

[29] J. Seo and B. Shneiderman. A rank-by-feature framework for unsupervised multidimensional data exploration using low dimensional projections. In *IEEE INFOVIS*, pages 65–72, 2004.

[30] M. Soltanolkotabi, E. Elhamifar, and E. J. Cands. Robust subspace clustering. *The Annals of Statistics*, 42(2):669–699, 2014.

[31] J. Stahnke, M. Dörk, B. Müller, and A. Thom. Probing projections: Interaction techniques for interpreting arrangements and errors of dimensionality reductions. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):629–638, 2016.

[32] A. Tatu, G. Albuquerque, M. Eisemann, J. Schneidewind, H. Theisel, M. Magnork, and D. Keim. Combining automated analysis and visualization techniques for effective exploration of high-dimensional data. In *2009 IEEE Symposium on Visual Analytics Science and Technology*, pages 59–66, 2009.

[33] A. Tatu, F. Maas, I. Farber, E. Bertini, T. Schreck, T. Seidl, and D. Keim. Subspace search and visualization to make sense of alternative clusterings in high-dimensional data. In *IEEE VAST*, pages 63–72, 2012.

[34] J. B. Tenenbaum, V. d. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[35] R. Tron and R. Vidal. A benchmark for the comparison of 3-d motion segmentation algorithms. In *IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[36] P. Tseng. Nearest q-flat to mpoints. *J. Optim. Theory Appl.*, 105(1):249–252, 2000.

[37] J. W. Tukey. Exploratory data analysis, 1977.

[38] L. Van Der Maaten. Accelerating t-sne using tree-based algorithms. *J. Mach. Learn. Res.*, 15(1):3221–3245, 2014.

[39] L. Van der Maaten and G. Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.

[40] R. Vidal. Subspace clustering. *IEEE Signal Processing Magazine*, 28(2):52–68, 2011.

[41] K. Wongsuphasawat, D. Moritz, A. Anand, J. Mackinlay, B. Howe, and J. Heer. Voyager: Exploratory analysis via faceted browsing of visualization recommendations. *IEEE Transactions on Visualization and Computer Graphics*, 22(1):649–658, 2016.

[42] J. Xia, W. Chen, Y. Hou, W. Hu, X. Huang, and D. S. Ebert. Dimscanner: A relation-based visual exploration approach towards data dimension inspection. In *IEEE Symposium on Visual Analytics Science and Technology*, 2016.

[43] X. Yuan, D. Ren, Z. Wang, and C. Guo. Dimension projection matrix/tree: Interactive subspace visual exploration and analysis of high dimensional data. *IEEE Trans. Vis. Comput. Graph.*, 19(12):2625–2633, 2013.

[44] Z. Zhang and H. Zha. Principal manifolds and nonlinear dimensionality reduction via tangent space alignment. *SIAM J. Sci. Comput.*, 26(1):313–338, 2005.