Learning Low-Dimensional Latent Graph Structures: A Density Estimation Approach

Li Wang and Ren-cang Li

***Abstract*— We aim to automatically learn a latent graph structure in a low-dimensional space from high-dimensional, unsupervised data based on a unified density estimation framework for both feature extraction and feature selection, where the latent structure is considered as a compact and informative representation of the high-dimensional data. Based on this framework, two novel methods are proposed with very different but intuitive learning criteria from existing methods. The proposed feature extraction method can learn a set of embedded points in a low-dimensional space by naturally integrating the discriminative information of the input data with structure learning so that multiple disconnected embedding structures of data can be uncovered. The proposed feature selection method preserves the pairwise distances only on the optimal set of features and selects these features simultaneously. It not only obtains the optimal set of features but also learns both the structure and embeddings for visualization. Extensive experiments demonstrate that our proposed methods can achieve competitive quantitative (often better) results in terms of discriminant evaluation performance and are able to obtain the embeddings of smooth skeleton structures and select optimal features to unveil the correct graph structures of high-dimensional data sets.**

***Index Terms*— Density estimation, feature selection, structure learning, unsupervised dimensionality reduction.**

I. Introduction

With the advance of science and technology, data sets collected for various real-world problems usually are of high dimensionality, such as images in computer vision, microarray data in bioinformatics, and text documents in text mining. In general, high dimensionality poses great challenges, including the curse of dimensionality, degraded performance with noisy and irrelevant features, high computational costs, and storage requirements. Dimensionality reduction is one of the effective ways to alleviate these issues by reducing the number of features. Depending on whether the original features should be maintained, dimensionality reduction can be categorized into two types: feature extraction and feature selection. Feature extraction attempts to create a transformation of the input space into a low-dimensional space that can preserve most of the relevant information [1], whereas feature selection aims to select a subset of features from a large amount of original features for a compact and informative representation [2]. The two types of approaches are not unrelated since: 1) feature selection can be considered as a discrete representation of feature extraction [3] and 2) feature selection is generally used as a preprocessing step to remove irrelevant or noisy features from the real-world data before conducting feature extraction.

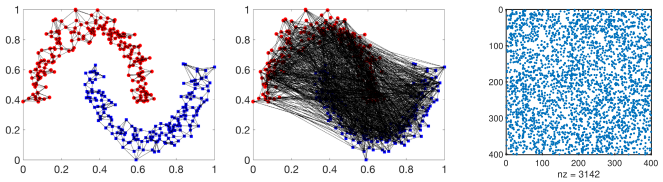


Fig. 1. Five-NN graph constructed from a synthetic two-moon data with two true features and 1000 augmented features sampled from a uniform distribution. (a) Five-NN graph on two true features. (b) Five-NN graph on all features visualized in the space of the two true features. (c) Adjacency matrix on the five-NN graph in (b), where *nz* stands for the number of nonzero elements.

In this paper, we are particularly interested in learning low-dimensional latent graph structures including feature extraction and feature selection for high-dimensional data under the unsupervised dimensionality reduction setting, where the label information is unavailable. Without the guidance of class labels, unsupervised dimensionality reduction is generally more challenging than supervised dimensionality reduction [4], let alone simultaneously learn a latent graph structure in a reduced space. It is worth noting that learning graph structures from data are inherently different from manifold (structure) learning, where the manifold is often approximated by a fixed graph. Here, our goal is to learn the graph structures by inferring them from data without any additional prior to the unknown structure. Due to the demand of various real-world applications, unsupervised dimensionality reduction has been continuously attracting tremendous attention, even though a large number of methods have been proposed (see [5]).

Without supervision, the manifold assumption has been widely employed as an important learning criterion in existing unsupervised dimensionality reduction methods [6], [7], but the structure directly computed from high-dimensional data such as k-nearest neighbor (NN) graph is not reliable. Given input data, its manifold structure is generally unknown and often approximations using precomputed neighborhood graphs are used by, e.g., feature extraction methods [6] and feature selection methods [7]. It is worth noting that these methods heavily rely on the correctness of pairwise distance/similarity matrices or graph structures precomputed from data with all

Manuscript received October 6, 2018; revised February 13, 2019; accepted May 14, 2019. Date of publication June 18, 2019; date of current version April 3, 2020. This work was supported in part by the NSF under Grant CCF-1527104 and Grant DMS-1719620. *(Corresponding author: Li Wang.)*

The authors are with the Department of Mathematics, University of Texas at Arlington, Arlington, TX 76019 USA (e-mail: li.wang@uta.edu; rcli@uta.edu).

This article has supplementary downloadable material available at http://ieeexplore.ieee.org, provided by the authors.

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TNNLS.2019.2917696

given features. To intuitively illustrate this issue, we show one synthetic example in Fig. 1 consisting of two true features and 1000 random noise features. The method [7] fails to find the two true features when the five-NN graph is constructed from all features. From Fig. 1(b) and (c), we can see that the five-NN graph built from all features is not informative to describe the underlying two-moon shape structure, so it is not reliable to approximate the true locality information as shown in Fig. 1(a) using all features with noise. Hence, it becomes important to trustfully reduce dimension (and hence suppress noise) in order to automatically learn a proper graph structure or similarity matrix from data.

In addition to modeling an appropriate structure, it is also important to take into account the discriminative information of data since feature extraction is performed not only for qualitative analysis such as data visualization but also for quantitative analysis such as classification and clustering. Although unlabeled data does not provide explicit discriminative information as *a priori*, t-distributed stochastic neighbor embedding (t-SNE) [8] is widely recognized due to its good discriminative performance and visualizing data with certain clustering/classification structures. A complementary approach called the maximum posterior manifold embedding (MPME) [9], [10] was proposed for inferring smooth skeleton structures from noisy data. It follows the similar criterion as the maximum variance unfolding (MVU) [11], [12] by preserving pairwise distance but formulates a probabilistic framework in order to model the data noise. Examples of real-world data sets illustrating the key difference of these methods can be found in Fig. 5. Without the guidance of discriminative information, MPME cannot achieve competitive discriminative results as t-SNE.

On the other hand, it is important for some applications to select a subset of features from the input data for either improving the learning performance by removing noisy or irrelevant features, or for obtaining a better interpretation of the given problem. Many studies [13], [14] for feature selection problems focus on the performance of classification and clustering. However, for some data sets, the underlying structure can be much more interesting than classification or clustering. As shown in Fig. 2, it is interesting to find the structure of a 2-D spiral shape embedded in the 3-D space. The special shape becomes more important than any clustering since no clustering pattern exists and distances on 2-D spiral shape make much more sense. More real-world examples will be studied in Section V. As discussed earlier, the manifold estimated from all features employed by above-mentioned methods may lose the true structure as shown in Fig. 1.

In order to automatically uncover the graph structure from data as a compact and informative representation of high-dimensional data, we aim to propose a unified density estimation approach for both feature extraction and feature selection in the unsupervised learning setting. Our motivation is to model the data generation process in that the "true" data are treated as a random variable following an unknown distribution, from which each point is drawn and corresponds to one observed data point. Based on this motivation, we model feature extraction and feature selection in a unified density estimation framework. Specifically, feature extraction models the embedded points randomly drawn from the "true" distribution, while feature selection can be achieved by assuming the "true" distribution depending only on the desirable set of features. Meanwhile, structure learning is then naturally encoded by constraints that preserve pairwise distances of any two "true" data points and the pairwise distances of the corresponding observed data points. Hence, our goal is to estimate the unknown density function, learn a graph structure, and obtain either embedded points for feature extraction or the desirable set of features for feature selection, simultaneously, in a unified framework. The main contributions of this paper are as follows.

1) We present a unified density estimation framework for feature extraction and feature selection. Our framework not only estimates the optimal density function over low-dimensional latent points but also automatically learns a graph structure over the low-dimensional embedding random variables for a compact and informative representation of high-dimensional data.

2) A new unsupervised feature extraction model is instantiated from the proposed framework with discriminant constraints. These constraints are derived from an approximate solution of the objective of t-SNE. Inherited from the property of the unified framework, our model also takes nonlinear similarity of high-dimensional data, discriminative clustering information, and less parameter tuning into account. These new properties resolve the issues that MPME suffers. Moreover, a new encoding process from the estimated density function is proposed for the embeddings of multiple disconnected components.

3) A novel unsupervised feature selection model is instantiated from the proposed framework with the ability to automatically weight the features of the input data in a high-dimensional space. Our derived objective function is built on three novel learning criteria. The resulting density is a function of a weighted graph and the optimal set of features, so it is independent of the high-dimensional data. Moreover, the low-dimensional embedding on the selected features can be easily obtained from the learned density function for the visualization of the learned graph structure.

4) Extensive experiments are conducted for evaluating the proposed unsupervised feature extraction method and unsupervised feature selection method by comparing with the state-of-the-art methods on a variety of data sets, including synthetic data sets and real-world applications. Our methods not only achieve discriminative embeddings and the optimal set of features but also successfully recover the smooth skeleton structures in a low-dimensional space.

II. Related Work

A large number of feature extraction techniques have been proposed during the last two decades [15], [16]. Most of them aim to preserve certain information of data. Principal component analysis (PCA) [17] minimizes the reconstruction error for learning a subspace linearly spanned by some orthonormal basis [16]. To achieve nonlinear transformation, kernel PCA (KPCA) [18] performs PCA in the reproducing kernel Hilbert space. Laplacian eigenmap (LE) [6] finds a mapping that minimizes the distances between a data point

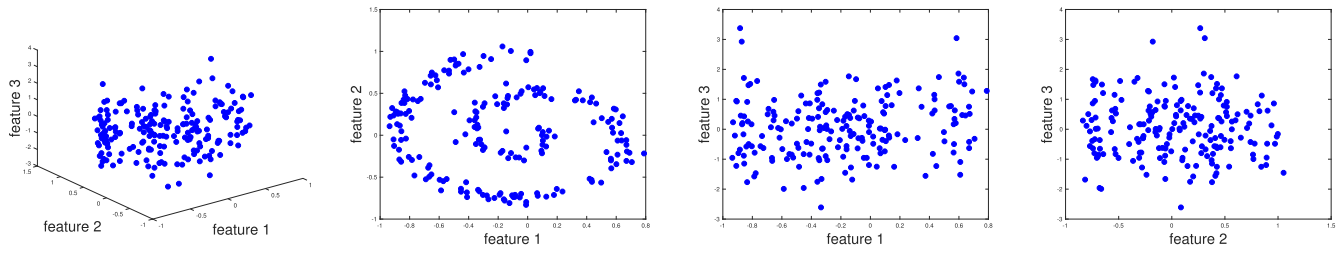


Fig. 2. Illustrated data with spiral structure over a subset of features. (a) Spiral data of features 1 and 2 with one random noisy feature 3. (b) Clearer structure in 2-D space than (c) and (d). (a) Spiral data in 3-D. (b) 2-D with features 1 and 2. (c) 2-D with features 1 and 3. (d) 2-D with features 2 and 3.

and its neighbors. Locally linear embedding (LLE) [19] preserves local geometry based on the assumption that local patches over K-NN are nearly linear and overlap with one another to form a manifold. Local tangent space analysis (LTSA) [20] describes local properties of the high-dimensional data using the local tangent space of each data point. t-SNE [8] employs a heavy-tailed distribution in the low-dimensional space to alleviate both the crowding problem and the difficulty of solving the optimization problem of SNE [21]. The performance of these methods critically depends on either the choice of the kernel function or a neighborhood graph.

Several works have been proposed for model selection of kernel functions. The Gaussian process latent variable model (GPLVM) [22] can achieve a nonlinear generalization of probabilistic PCA (PPCA) [23] and learn a kernel function defined on a set of variables in a low-dimensional latent space, but the objective function of GPLVM is nonconvex and the optimization problem is difficult to solve. MVU [11] learns a nonparametric kernel matrix by retaining pairwise distances encoded in a neighborhood graph constructed from the input data, but it is impractical to scale up to moderate size. Landmark MVU (ℓMVU) [12] alleviates the high computational complexity of MVU by introducing landmarks and a linear transformation for the kernel matrix factorization based on various assumptions. Maximum entropy unfolding (MEU) [24] directly models the density of observed data and embedded points are obtained by maximizing the likelihood of the learned density.

Structure learning has had a great success in automatically constructing structures of data for learning proper embeddings. A sparse manifold clustering and embedding (SMCE) [25] is proposed to measure the linear representation of every data point by using its neighborhood information. ℓ_1 graph is learned for enhancing the robustness of the learned graph [26]. In addition to directed graphs, an integrated model for learning an undirected graph is proposed [27]. These are deterministic models, and they lack the ability to handle the noise of data. To tackle noisy data and achieve smooth skeleton structures, MPME [9], [10] was recently proposed based on the distance preservation, but it lacks the guidance of discriminative information.

On the other hand, existing unsupervised feature selection methods rely on various types of assumptions as of the learning criteria. The widely used criterion is to score each feature according to certain manifold structures based on a graph constructed from all features. Typical methods are Laplacian score (LS) [7], spectral feature selection (SPEC) [28], and multiclass feature selection (MCFS) [29]. It is intuitive to jointly formulate unsupervised feature selection based on assumptions of both clustering and manifold so that both criteria can interact with each other. The representative methods include: joint embedding learning and spectral regression (JELSR) [13], nonnegative discriminative feature selection (NDFS) [30], robust unsupervised feature selection (RUFS) [31], robust SPEC (RSFS) [32], and feature selection via clustering-guided sparse structural learning (CGSSL) [14]. Among these methods, the clustering assumption is formulated by either regression problem or matrix factorization, and features are selected by using the $\ell_{2,1}$ regularizer over the coefficients of a linear regression model.

Another criterion for unsupervised feature selection is to preserve the pairwise similarity between two data points in the original feature space. Similarity preserving feature selection (SPFS) [33] proposes to select features based on the assumption that the pairwise similarities with these features can be maximally preserved, where a pairwise similarity matrix is precomputed from the original data. Global and local structure preservation for feature selection (GLSPFS) [34] extends SPFS by incorporating the local manifold structure of data. Nonlinear joint unsupervised feature selection (NJUFS) [35] maximizes the alignment between the pairwise similarity matrix from all features and the similarity matrix from a subset of features in terms of the Hilbert–Schmidt independence criterion [36]. Stochastic neighbor-preserving feature selection (SNFS) [37] selects the features that can best preserve stochastic neighbors.

Structure learning has also been explored for unsupervised feature selection. Local learning-based clustering feature selection (LLCFS) [38] models feature selection within the framework of the local learning-based clustering method where the induced graph Laplacian can be iteratively updated. Structured optimal graph feature selection (SOGFS) [39] learns a similarity matrix based on the assumption that closer samples are likely to connect with a larger probability. Feature selection with adaptive structure learning (FSASL) [5] preserves the sparse reconstruction structure and local manifold structure. Both methods [5], [39] employ an explicit discriminative projection for given classes. This may weaken the power of structure learning due to the strict clustering assumption that data points from different clusters should be distant. Preserving pairwise similarity [33] or stochastic neighbors [37] can be a

good complement for structure learning, but they did not aim to learn latent graph structures.

III. Unsupervised Feature Extraction via Density Estimation

We propose to estimate the density function of embedded points by regulating the pairwise distance between any two embedded points so that both clustering and manifold structure of the original data can be properly captured. Once the density function is obtained from data, the embedded points are then uncovered well for both clusterings and learning manifold structures.

A. Motivation

Let $\mathcal{D} = \{\mathbf{x}_1, \ldots, \mathbf{x}_N\}$ be an M-dimensional input data set with $\mathbf{x}_i \in \mathbb{R}^M, \forall i$, and $d(\mathbf{x}_i, \mathbf{x}_j)$ be a distance function between two points $\mathbf{x}_i$ and $\mathbf{x}_j$, e.g., the Euclidean distance $d(\mathbf{x}_i, \mathbf{x}_j) = ||\mathbf{x}_i - \mathbf{x}_j||^2$. Our goal is to learn an m-dimensional embedding $\mathcal{D}_m = \{\mathbf{y}_1, \ldots, \mathbf{y}_N\}$ so that $\mathbf{x}_i \in \mathbb{R}^M$ is represented by a point $\mathbf{y}_i \in \mathbb{R}^m$ with $m \leq M$.

We seek dimensionality reduction methods that can seamlessly integrate two properties, i.e., discriminative information and manifold structure, encoded in the high-dimensional input data. The distance function is one of the important statistics to measure the two properties, but most dimensionality reduction methods only focus on one of the two factors. Next, we show that SNE and t-SNE concentrate on the clustering preservation of the embedded points.

We revisit SNE [21] and t-SNE [8] to give an explanation on their good clustering performance. SNE defines conditional probabilities $p_{j|i}$ to measure the pairwise similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$ given by

$$p_{j|i} = \frac{\exp\left(-d(\mathbf{x}_i, \mathbf{x}_j)/2\sigma_i^2\right)}{\sum_{k \neq i} \exp\left(-d(\mathbf{x}_i, \mathbf{x}_k)/2\sigma_i^2\right)} \quad \forall j \neq i, \;\; p_{i|i} = 0 \tag{1}$$

where σ_i is the bandwidth of the Gaussian kernels and it is set in such a way that the perplexity of the conditional distribution equals to a predefined perplexity u. t-SNE defines a joint probabilities $p_{i,j}$ by symmetrizing two conditional probabilities given by

$$p_{i,j} = \frac{p_{j|i} + p_{i|j}}{2N} \quad \forall i \neq j. \tag{2}$$

In the m-dimensional embedding $\mathcal{D}_m$, the similarity between two points $\mathbf{y}_i$ and $\mathbf{y}_j$ are defined similarly. SNE takes the same functions as the conditional probabilities (1) by fixing the bandwidth $2\sigma_i^2 = 1, \forall i$, as

$$q_{j|i} = \frac{\exp(-d(\mathbf{y}_i, \mathbf{y}_j))}{\sum_{k \neq i} \exp(-d(\mathbf{y}_i, \mathbf{y}_k))} \quad \forall j \neq i, \;\; q_{i|i} = 0 \tag{3}$$

while t-SNE chooses a normalized Student-t kernel function with a single degree of freedom as

$$q_{i,j} = \frac{(1 + d(\mathbf{y}_i, \mathbf{y}_j))^{-1}}{\sum_{k \neq l} (1 + d(\mathbf{y}_k, \mathbf{y}_l))^{-1}} \quad \forall j \neq i, \;\; q_{i,i} = 0 \tag{4}$$

which is able to accurately model small pairwise distances in the low-dimensional space.

Finally, the embedding $\mathcal{D}_m$ is obtained by minimizing the Kullback–Leibler (KL) divergence between two conditional or joint probabilities for SNE or t-SNE, respectively, e.g., SNE solves the following optimization problem:

$$\mathcal{D}_m = \arg\min \mathrm{KL}(P||Q) = \sum_{i \neq j} p_{j|i} \log \frac{p_{j|i}}{q_{j|i}}. \tag{5}$$

Problem (5) is then solved by the gradient descent method with the Euclidean distance function.

The clustering interpretation is motivated from the special structure of the conditional probabilities in (1). We first construct an optimization problem with its optimal solution equivalent to (1) and then infer the clustering property by analyzing the constructed objective function.

Proposition 1 gives a desired constructed function. Next, we will illustrate the fact that (6) helps SNE and t-SNE to achieve better clustering performance on the embeddings.

Proposition 1: Problems

$$\min_{\{p_{j|i}\} \in \mathcal{P}_i} \sum_{j \neq i} p_{j|i}(d(\mathbf{x}_i, \mathbf{x}_j) + 2\sigma^2 \log p_{j|i}) \quad \forall i \tag{6}$$

have the optimal solution given by (1), where

$$\mathcal{P}_i = \left\{ p_{j|i} \Big| \sum_{j \neq i} p_{j|i} = 1, \, p_{j|i} \geq 0, \forall j \neq i \right\} \quad \forall i.$$

Proof: This can be proved by applying the Lagrangian duality theorem [40] to (6) with constraints, and the primal variable has the analytic solution (1). □

Problem (6) is the same as the membership assignment of the possibilistic c-means (PCM) [41] if $p_{j|i}$ is interpreted as the probability of assigning $\mathbf{x}_j$ to membership $\mathbf{x}_i$. The differences are that: 1) the number of centroids equals N and 2) the cluster centroids are fixed. Specifically, the PCM solves the following joint optimization problem:

$$\min_{\{\mathbf{c}_i\}} \min_{\{p_{j|i}\} \in \mathcal{P}_i} \sum_{j \neq i} p_{j|i}(d(\mathbf{c}_i, \mathbf{x}_j) + 2\sigma^2 \log p_{j|i}) \quad \forall i \tag{7}$$

where $\{\mathbf{c}_i\}_{i=1}^N$ is a set of centroids to optimize. Accordingly, $p_{j|i}$ is a pairwise similarity that encodes the cluster assignment between a data point and its cluster centroid in the input space.

SNE applies the same similarity function over $\mathcal{D}_m$ with $2\sigma_i^2 = 1, \forall i$, by solving, $\forall i$

$$\min_{\{q_{j|i}\} \in \mathcal{Q}_i} \sum_{j \neq i} q_{j|i}(d(\mathbf{y}_i, \mathbf{y}_j) + \log q_{j|i}) \tag{8}$$

$$\text{s.t.} \mathcal{Q}_i = \left\{ q_{j|i} \Big| \sum_{j \neq i} q_{j|i} = 1, q_{j|i} \geq 0, \forall j \neq i \right\}$$

which has the optimal solution that equals the conditional probability of the embeddings in (3).

The learning criterion to obtain the optimal embedding by SNE is to maximally maintain the clustering assignment matrix (i.e., the pairwise similarities) from $p_{j|i}$ to $q_{j|i}$ as much as possible. This can be achieved by updating embedded points

to minimize the difference between two distributions in (5). As a result, the clustering structures are preserved from the original data to embedded points.

For t-SNE, we have the similarly constructed optimization problem as shown in Proposition 2 for the conditional probabilities of the embedded points. This corresponds to a variant of PCM [42], so that the above interpretation as clustering assignment to SNE can also be applied to t-SNE.

Proposition 2: Problem

$$\min_{\{q_{i,j}\}\in\mathcal{Q}} \sum_{j\neq i} \left(q_{i,j}^2 d(\mathbf{y}_i, \mathbf{y}_j) + (1 - q_{i,j})^2\right), \tag{9}$$

has the optimal solution given by (4), where

$$\mathcal{Q} = \left\{ q_{i,j} \middle| \sum_{j\neq i} q_{i,j} = 1, q_{i,j} \geq 0, \forall i \neq j \right\}. \tag{10}$$

Therefore, both SNE and t-SNE treat cluster assignment probabilities as the most important information to preserve for learning the low-dimensional embedded points. The constructed optimization problems (6), (8), and (9) inspires the proposed method here to capture the discriminative information of the input data using the pairwise distance constraints, which is discussed in Section III-B as the key component of the proposed feature extraction method.

B. Proposed Unsupervised Feature Extraction Method

We propose to formulate the dimensionality reduction problem as a unified model in order to naturally integrate both clustering and manifold structure learning. To achieve this goal, our intuition is to transform the constructed optimization problems of (t-)SNE to a set of pairwise distance constraints for graph structure learning.

Let $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_N] \in \mathbb{R}^{m\times N}$ be a set of embedded points with $\mathbf{y}_i \in \mathbb{R}^m$. Assume that the unknown density function is $p(\mathbf{Y})$, where the density functions $p(\mathbf{y}^r)$ with respect to rows $\mathbf{y}^r, \forall r$, are independent

$$p(\mathbf{Y}) = \prod_{r=1}^{m} p(\mathbf{y}^r), \quad \pi(\mathbf{Y}) = \prod_{r=1}^{m} \pi(\mathbf{y}^r) \tag{11}$$

and the prior distribution π over $\mathbf{Y}$ has the same assumption. Since $\mathbf{Y}$ is a matrix of random variables, the pairwise Euclidean distance between $\mathbf{y}_i$ and $\mathbf{y}_j$ is defined as the expectation over $p(\mathbf{Y})$ given by

$$\mathbb{E}_{p(\mathbf{Y})}[||\mathbf{y}_i - \mathbf{y}_j||^2] = \sum_{r=1}^{m} \int_{\mathbf{y}^r} (y_{r,i} - y_{r,j})^2 \; p(\mathbf{y}^r) \mathrm{d}\mathbf{y}^r \tag{12}$$

where the equality holds due to (11).

According to the discussion in Section III-A, the clustering information of embedded points are encoded within the constructed problem

$$\min_{\{q_{i,j}\}\in\mathcal{Q}} \sum_{j\neq i} (q_{i,j}(d(\mathbf{y}_i, \mathbf{y}_j) + \lambda \log q_{i,j})) \tag{13}$$

where λ is a regularization parameter and its optimal $q_{i,j}$ is given by the following equalities:

$$q_{i,j} = \frac{\exp(-d(\mathbf{y}_i, \mathbf{y}_j)/\lambda)}{\sum_{k\neq l} \exp(-d(\mathbf{y}_l, \mathbf{y}_k)/\lambda)} \quad \forall j \neq i. \tag{14}$$

If we preserve the cluster assignment matrix computed from the input data, i.e., $q_{i,j} = (p_{j|i} + p_{i|j})/(2N), \forall i, j$, we have

$$\frac{p_{j|i} + p_{i|j}}{2N} = \frac{1}{z} \exp(-d(\mathbf{y}_i, \mathbf{y}_j)/\lambda) \quad \forall i, j \tag{15}$$

where $z = \sum_{k\neq l} \exp(-d(\mathbf{y}_l, \mathbf{y}_k)/\lambda)$ is the normalization term and $p_{j|i}$ is computed using (1). To prevent the pairwise distances from arbitrary scaling, we impose $z = N$ as a normalization term. The Taylor expansion on the right-hand side of (15) can be used to obtain the following equality for each pairwise distance

$$\begin{aligned} \frac{p_{j|i} + p_{i|j}}{2} &= \exp(-d(\mathbf{y}_i, \mathbf{y}_j)/\lambda) \\ &= \sum_{n=0}^{\infty} \frac{(-d(\mathbf{y}_i, \mathbf{y}_j)/\lambda)^n}{n!}, \forall i, j. \end{aligned}$$

By imposing $d(\mathbf{y}_i, \mathbf{y}_j)/\lambda \in [0, 1]$, we have $(p_{j|i} + p_{i|j})/2 \leq 1 - d(\mathbf{y}_i, \mathbf{y}_j)/\lambda$, so the following inequality always holds:

$$d(\mathbf{y}_i, \mathbf{y}_j) \leq \lambda \left(1 - \frac{p_{j|i} + p_{i|j}}{2}\right) \quad \forall i, j. \tag{16}$$

As $p_{j|i}$ encodes the clustering assignment information of the input data, it is conceived that the constraints (16) take clustering information into account. This leads to the key difference from the distance preservation methods such as MVU and MPME, where the Euclidean distance is modeled either directly or indirectly via a kernel function such that

$$||\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)||^2 = \kappa(\mathbf{x}_i, \mathbf{x}_i) + \kappa(\mathbf{x}_j, \mathbf{x}_j) - 2\kappa(\mathbf{x}_i, \mathbf{x}_j) \tag{17}$$

where κ is a kernel function, e.g., the Gaussian kernel with bandwidth σ, given by

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \exp(-||\mathbf{x}_i - \mathbf{x}_j||^2/2\sigma^2). \tag{18}$$

We have $||\phi(\mathbf{x}_i) - \phi(\mathbf{x}_j)||^2 = 2(1 - \kappa(\mathbf{x}_i, \mathbf{x}_j))$, where the relations among input points can only capture the relationship between $\mathbf{x}_i$ and $\mathbf{x}_j$, so there is no clustering information encoded. Moreover, the bandwidth is dynamically varied for each input data according to the density of the data in (16), while distance preservation methods (*e.g.*, MPME) employ a single fixed σ as a tuning parameter and MVU relies on a neighborhood graph with fixed neighbor size. By comparisons, the above analyses imply that (16) not only can capture the clustering information of the input data but also can automatically adjust the parameter of each point according to the density of the input data.

Moreover, the prior distribution of the proposed method is set as a multivariate normal distribution with mean zero and covariance as the identity matrix, i.e., $\pi(\mathbf{y}^r) \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. This strategy is similar to (t-)SNE where a fixed variance is used for $q_{j|i}$ or $q_{i,j}, \forall i, j$. Since $p(\mathbf{Y})$ is a density function to optimize, this prior can also constrain the density in a proper embedding space without tuning.

Based on the above definitions, we propose a new formulation for estimating the density of embedded points via

regularized Bayesian inference given by

$$\min_{p(\mathbf{Y}),\{\xi_{i,j}\}} \int p(\mathbf{Y}) \log \frac{p(\mathbf{Y})}{\pi(\mathbf{Y})} \mathrm{d}\mathbf{Y} + C \sum_{i,j} \xi_{i,j} \quad (19)$$
$$\text{s.t.}\, \mathbb{E}_{p(\mathbf{Y})}[||\mathbf{y}_i - \mathbf{y}_j||^2] \le \lambda \left(1 - \frac{p_{j|i} + p_{i|j}}{2}\right) + \xi_{i,j}, \xi_{i,j} \ge 0 \quad \forall i, j$$
$$\int p(\mathbf{y}^r) \mathrm{d}\mathbf{y}^r = 1 \quad \forall r$$

where λ is a parameter for controlling the clustering information.

Both problem (19) and MPME estimate the density of embedded points via regularized Bayesian inference to incorporate pairwise distance constraints, but there are three key differences as follows.

1) Nonlinear similarities are used in (19), i.e., problem (19) can be treated as a nonlinear extension of MPME.
2) The clustering information is encoded in the constraints.
3) The prior distribution is a multivariate normal distribution with no tuning parameter for the variance.

Both problem (19) and t-SNE take the conditional probabilities as the input for learning embedded points, but they are very different approaches. First, problem (19) estimates a continuous density function of the embedded points, while t-SNE models the discrete embedded points. Hence, the KL-divergence criteria are applied in a different way. Second, problem (19) not only takes conditional probabilities into account but also preserves the pairwise distances. Third, problem (19) automatically learns a smooth skeleton structure from data and also is convex so its global solution can be achieved, while t-SNE does not share these desirable properties.

C. Optimization Algorithm

Problem (19) is jointly convex with respect to $p(\mathbf{Y})$ and $\{\xi_{i,j}\}$. We obtain its dual problem by the Lagrangian duality [40], which is summarized in Proposition 3.

Proposition 3: Given $\pi(\mathbf{y}^r) \sim \mathcal{N}(\mathbf{0}, \mathbf{I}), \forall r$ and (11), the dual problem of (19) is the following optimization problem

$$\min_{\mathbf{W}} \; -\frac{m}{2} \log\det(\mathbf{I} + 4\mathbf{L}) + \lambda \sum_{i,j} w_{i,j} \left(1 - \frac{p_{j|i} + p_{i|j}}{2}\right)$$
$$\text{s.t. } 0 \le w_{i,j} \le C \quad \forall i, j \quad (20)$$

where the (i, j)th entry $w_{i,j}$ of multiplier matrix $\mathbf{W} \in \mathbb{R}_+^{N \times N}$ is introduced for the (i, j)th constraint, $\mathbf{L} = \text{diag}(\mathbf{W}\mathbf{1}) - \mathbf{W}$ with $\mathbf{1}$ being the column vector of all ones. The estimated density function is

$$p(\mathbf{y}^r) \propto \pi(\mathbf{y}^r) \exp\left(-\sum_{i,j} w_{i,j} (y_{r,i} - y_{r,j})^2\right) \quad \forall r. \quad (21)$$

The problem (20) is convex, so any general nonlinear optimization method can achieve its global solution. Even for a large number of variables, problem (20) can be efficiently solved by the L-BFGS-B algorithm [43].

Let $\mathbf{Q} = \mathbf{I} + 4\mathbf{L}$ and the objective function of (20) be

$$f(\mathbf{W}) = -\frac{m}{2} \log\det(\mathbf{Q}) + \lambda \sum_{i,j} w_{i,j} \left(1 - \frac{p_{j|i} + p_{i|j}}{2}\right). \quad (22)$$

The gradient of $f(\mathbf{W})$ with respect to $w_{i,j}$ is

$$\partial_{w_{i,j}} f(\mathbf{W}) = \lambda \left(1 - \frac{p_{j|i} + p_{i|j}}{2}\right) - 2\, m\text{Tr}(\mathbf{Q}^{-T} \mathbf{A}_{i,j}) \quad (23)$$

$$\mathbf{A}_{i,j}(s,t) = \begin{cases} -1, & s \ne t \wedge s = i \wedge t = j \\ 1, & s = t = i \ne j \\ 0, & \text{otherwise.} \end{cases} \quad (24)$$

Since $\mathbf{A}_{i,j}$ has only two nonzero entries, the gradient can be easily computed as, $\forall i, j$

$$\partial_{w_{i,j}} f(\mathbf{W}) = \lambda \left(1 - \frac{p_{j|i} + p_{i|j}}{2}\right) - 2m(\mathbf{Q}^{-1}(i,i) - \mathbf{Q}^{-1}(i,j)).$$

The gradient values satisfy the symmetric property, i.e., $\partial_{w_{i,j}} f(\mathbf{W}) = \partial_{w_{j,i}} f(\mathbf{W})$ since both $\mathbf{Q}^{-1}$ and $(1 - (p_{j|i} + p_{i|j})/2)$ are symmetric. $w_{i,i} = 0, \forall i$ for constraints that are always true since the distance between a point and itself is zero.

Problem (20) has a trivial solution, i.e., $\mathbf{W} = \mathbf{0}$ if λ is set improperly. It is clear that $f(\mathbf{0}) = 0$. To prevent the trivial solution, we require a λ such that

$$f(\mathbf{W}) < 0. \quad (25)$$

However, it is hard to directly estimate from the above-mentioned inequality when the optimal solution is unknown. We can roughly estimate λ in the first iteration of the L-BFGS-B by imposing

$$\exists (i,j), \partial_{w_{i,j}} f(\mathbf{W}) < 0 \Rightarrow \lambda < \lambda^u = \min_{i,j} \frac{2m}{1 - \frac{p_{j|i} + p_{i|j}}{2}} \quad (26)$$

which guarantees that the first iteration starting from initial $\mathbf{W} = \mathbf{0}$ will decrease to a nontrivial solution in the gradient descent direction. For the ease of tuning parameter λ, we set $\lambda = \lambda^v \lambda^u$ with λ^v varying in $(0, 1)$.

D. Embeddings of Multiple Disconnected Components

The embedding $\mathbf{Y}$ is then decoded from the learned $\mathbf{W}$. According to (21), we have the estimated density, $\forall r$

$$p(\mathbf{y}^r) = \frac{\pi(\mathbf{y}^r) \exp(-\sum_{i,j} w_{i,j} (y_{r,i} - y_{r,j})^2)}{\int \pi(\mathbf{y}^r) \exp(-\sum_{i,j} w_{i,j} (y_{r,i} - y_{r,j})^2) \mathrm{d}\mathbf{y}^r} \sim \mathcal{N}(\mathbf{y}^r | \mathbf{0}, \mathbf{Q}^{-1}) \quad (27)$$

where $\mathbf{Q} = \mathbf{I} + 4\mathbf{L}$. As a result, we have the expectation of the pairwise distance between latent variables $\mathbf{y}_i$ and $\mathbf{y}_j$ as

$$\mathbb{E}_{p(\mathbf{Y})}[||\mathbf{y}_i - \mathbf{y}_j||^2] = m[\mathbf{Q}^{-1}(i,i) + \mathbf{Q}^{-1}(j,j) - 2\mathbf{Q}^{-1}(i,j)] \quad \forall i, j.$$

To explicitly represent $\mathbf{y}_i$, we take the point estimate as $\widehat{\mathbf{y}}_i$, such that

$$||\widehat{\mathbf{y}}_i - \widehat{\mathbf{y}}_j||^2 = m[\mathbf{Q}^{-1}(i,i) + \mathbf{Q}^{-1}(j,j) - 2\mathbf{Q}^{-1}(i,j)] \quad \forall i, j$$

which are equivalent to the following equalities:

$$\widehat{\mathbf{y}}_i^T \widehat{\mathbf{y}}_j = m\mathbf{Q}^{-1}(i,j) \quad \forall i, j. \tag{28}$$

As a result, $\mathbf{Q}^{-1}$ is a valid kernel for the latent variables $\mathbf{Y}$ since Laplacian matrix $\mathbf{L}$ is semidefinite and $\mathbf{Q}$ is positive definite. Similar to KPCA, kernel centralization is required by setting $\widehat{\mathbf{K}} = (\mathbf{I} - (1/N)\mathbf{1}\mathbf{1}^T)\mathbf{Q}^{-1}(\mathbf{I} - (1/N)\mathbf{1}\mathbf{1}^T)$, which is to make $\mathbf{y}_i$ centered in the feature space. Hence, we can obtain $\mathbf{Y}$ by performing eigendecomposition on $\widehat{\mathbf{K}}$ as $\widehat{\mathbf{K}} = \mathbf{U}\mathbf{V}\mathbf{U}^T$ and $\mathbf{Y} = \mathbf{U}\mathbf{V}^{1/2}$ by keeping the top m bases with the largest eigenvalues.

It is well-known that the Laplacian matrix has zero as eigenvalues, the number of which corresponds to the number of disconnected graph components represented by the adjacency matrix $\mathbf{W}$. The eigenspace associated with eigenvalue zero is spanned by the indicator vectors of those components [44]. Hence, we can break down the similarity matrix $\mathbf{W}$ into submatrices, each of which corresponds to one connected component. For each component, we can separately apply the above decoding process for the embedded points, which only corresponds to this component. As a result, we can obtain multiple sets of embedded points corresponding to multiple disconnected components of the learned graph structure. However, t-SNE does not have this property.

E. Discussion

The proposed model (20) has several interesting properties. The objective function contains log-determinant of $\mathbf{I} + 4\mathbf{L}$, which can be equivalently formulated as $\log\det(\mathbf{I} + 4\mathbf{L}) = N\log 4 + \sum_{i=1}^{N}\log(\gamma_i + 1/4)$, where γ_i denotes the ith largest eigenvalue of the symmetric matrix $\mathbf{L}$. Thus, the log-determinant can be related to the negative log-likelihood of a power law distribution of γ_i as $p(\gamma_i) \propto \gamma_i^{-\theta}$ where θ is called the power law exponent. The power law distribution imposes large values on a small set of eigenvalues, while small values on the rest of eigenvalues. Hence, $\theta = (m/2)$ is used in the proposed model. This is critically different from the dual MVU problem where the second smallest eigenvalue is maximized [45]. Since the power law distribution prefers a small number of large eigenvalues and a large number of small eigenvalues, the learned similarity matrix leads to a smooth skeletonlike structure as discussed in [9] and [10].

The proposed model (19) takes the discriminative information similar to (t-)SNE into account, but a crucially different optimization criterion is employed. (t-)SNE directly preserves the clustering assignment matrix as shown in (5), while our model learns the underlying manifold structure by incorporating the clustering assignment matrix into pairwise distance constraints. This is the key to generate very different embedded points from (t-)SNE, although the same joint probabilities are used as the input.

The pseudocode of our proposed embedding via structure learning (ESL) is given in Algorithm 1. Solving problem (20) takes approximately $O(N^{2.37})$ for computing logdet and inverting $\mathbf{Q}$ at each iteration of L-BFGS-B solver. Performing embedding via KPCA takes $O(N^3)$. Thus, the time complexity of Algorithm 1 takes the order of $O(N^3)$, which is the same as most spectral-based methods but is much faster than semidefinite programming used in MVU. The computational complexity is the same as MPME with an additional cost for computing the joint probabilities as in t-SNE. For simplicity, λ^v is renamed as λ in the experiments.

Algorithm 1 Embedding via Structure Learning

1: **Input**: data $\mathbf{X}$, perplexity u, parameters $\lambda^v \in (0, 1)$ and C
2: compute pairwise affinities $p_{j|i}$ with perplexity u
3: set $\lambda = \lambda^v \lambda^u$ using (26)
4: obtain $\mathbf{W}$ by solving convex problem (20) using L-BFGS-B
5: detect the disconnected components of $\mathbf{W}$
6: perform embedding on each component
7: **Output:** a list of sets of embedded points

IV. Unsupervised Feature Selection via Density Estimation

We propose to estimate the density function of data points with respect to the set of selected features by preserving the pairwise distances only on the optimal set of features and learning the structure of data, simultaneously.

A. Motivation

Given the observed data $\mathbf{X}$ in an M-dimensional space, we aim to select a subset of features so that the pairwise distances are preserved only in terms of the selected features, and meanwhile, the distribution of the data with the selected features should differ from a distribution for which feature weights are equally important.

To preserve the pairwise distances, we first define the following parameterized pairwise distance function, for unsupervised feature selection, given by:

$$\psi(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}) = \sum_{r=1}^{M} \theta_r (x_{r,i} - x_{r,j})^2 \tag{29}$$

where $\boldsymbol{\theta} = [\theta_1, \ldots, \theta_M]^T \in \mathbb{R}_+^M$ is a column vector for the importance of features. For distance-based methods, it is clear that if $\theta_r > 0$, the rth feature is selected, and $\theta_r = 0$ stands for an unimportant feature. Moreover, the larger the value of θ_r is, the more important the rth feature will be. We define the feasible set of $\boldsymbol{\theta}$ as

$$\Theta = \left\{ \boldsymbol{\theta} \,\middle|\, \sum_{r=1}^{M} \theta_r = b, 0 \le \theta_r \le 1, \forall r \right\} \tag{30}$$

where b is a parameter that controls the number of selected features. This constraint has been used in supervised feature selection methods (see [46]).

Furthermore, a matrix $\mathbf{Y} \in \mathbb{R}^{M\times N}$ of latent random variables is introduced to generate the true data from an unknown true density function. The modeling of true distribution here

is different from feature extraction in Section III, where the true distribution is defined over the embedded points in an m-dimensional space. Denote the rth row of $\mathbf{Y}$ as $\mathbf{y}^r \in \mathbb{R}^N$ and the ith column of $\mathbf{Y}$ as $\mathbf{y}_i \in \mathbb{R}^M$. Suppose that $\mathbf{Y}$ follows some unknown distribution $p(\mathbf{Y})$ with the independent assumption on the rows of $\mathbf{Y}$, i.e., $p(\mathbf{Y}) = \prod_{r=1}^{M} p(\mathbf{y}^r)$. We further assume that $p(\mathbf{Y})$ is the true distribution that generates $\mathbf{X}$ with two types of noise: the noise contaminated the important features and the noise of newly added noisy features. Similarly, the pairwise distance between two true random variables $\mathbf{y}_i$ and $\mathbf{y}_j$ can be represented as the expected pairwise distance over the true distribution as

$$\mathbb{E}[\psi(\mathbf{y}_i, \mathbf{y}_j; \boldsymbol{\theta})] = \sum_{r=1}^{M} \theta_r \int_{\mathbf{y}^r} (y_{r,i} - y_{r,j})^2 \; p(\mathbf{y}^r) \mathrm{d}\mathbf{y}^r \quad (31)$$

where the equality holds due to the independent assumption of the rows of $\mathbf{Y}$ and the distance definition (29).

Let $\pi_0(\mathbf{y}^r) \sim \mathcal{N}(\mathbf{0}, \gamma^{-1}\mathbf{I}), \forall i$ and $\pi_0(\mathbf{Y}) = \prod_{r=1}^{M} \pi_0(\mathbf{y}^r)$ so that prior features are equally important, e.g., $\mathrm{diag}(\boldsymbol{\theta}) = \gamma \mathbf{I}$ as the precision matrix of prior distribution, where $\gamma > 0$ is interpreted as the parameter for noise. Three important criteria are summarized as follows.

Criterion 1 (Distance Preservation): The expected pairwise distance over the true distribution is maintained according to certain neighborhood structure and the proper $\boldsymbol{\theta}$, that is,

$$\mathbb{E}[\psi(\mathbf{y}_i, \mathbf{y}_j; \boldsymbol{\theta})] = \psi(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}) \quad \forall i, j. \quad (32)$$

The relationship between structure learning and noisy distance preservation will be clear in Section IV-B. If $\boldsymbol{\theta}$ is a vector of ones, the constraints are the same as these in MPME.

Criterion 2 (Maximum Deviation Criterion for $\boldsymbol{\theta}$): This criterion can be formulated as an optimization problem by maximizing the discrepancy between true distribution $p(\mathbf{Y}; \boldsymbol{\theta})$ and the prior $\pi_0(\mathbf{Y})$ with respect to $\boldsymbol{\theta} \in \Theta$, so that $p(\mathbf{Y}; \boldsymbol{\theta})$ is distant from random noisy prior $\pi_0(\mathbf{Y})$ and the informative structure of data is captured. Note that $p(\mathbf{Y}; \boldsymbol{\theta})$ is dependent on $\boldsymbol{\theta}$ when distance preservation is used.

Criterion 3 (Maximum Entropy Criterion for $p(\mathbf{Y})$): Without any prior information, the best criterion is the maximum entropy principle [47]. On the other hand, given a prior distribution $\pi_0(\mathbf{Y})$, it is better to minimize the distance between distributions $p(\mathbf{Y})$ and $\pi_0(\mathbf{Y})$ with respect to $p(\mathbf{Y})$ so that the true data are close to the prior without additional knowledge provided. This was used in the regularized Bayesian inference [48].

The above-mentioned three learning criteria are very different from the criteria used in the existing unsupervised feature selection methods [5], [7]. Next, with these criteria in mind, we formulate a novel unsupervised feature selection method.

B. Proposed Unsupervised Feature Selection Method

In this section, we propose a novel method for unsupervised feature selection and structure learning. Based on the above-mentioned three criteria, the joint framework for unsupervised feature selection and structure learning is formulated as the following maximin optimization problem

$$\max_{\boldsymbol{\theta} \in \Theta} \min_{p(\mathbf{Y}) \in \mathcal{P}, \{\rho_i\}_{i=1}^{N}} \mathrm{KL}(p(\mathbf{Y}) || \pi_0(\mathbf{Y})) + \lambda \sum_{i=1}^{N} \rho_i$$
$$\text{s.t. } \mathbb{E}[\psi(\mathbf{y}_i, \mathbf{y}_j; \boldsymbol{\theta})] \leq \psi(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}) + \rho_i \quad \forall i, j \quad (33)$$

where $\mathcal{P} = \{p(\mathbf{y}^r) | \int p(\mathbf{y}^r) \mathrm{d}\mathbf{y}^r = 1, p(\mathbf{y}^r) > 0, \forall r\}$, λ is a tradeoff parameter, and the inequality constraints used for nonnegative weighted graph learning. The KL-divergence and distance constraints in (33) directly follow Criteria 3 and 1, respectively. The slack variable ρ_i is maximized over $\boldsymbol{\theta}$ for feature selection because of the unreliable pairwise distance, so Criterion 2 is applied, and it is minimized for density function in order to maintain the distance given the true set of features, so Criterion 1 is applied. Hence, problem (33) is formulated to satisfy above-mentioned three criteria by optimizing the importance of features $\boldsymbol{\theta}$ and density $p(\mathbf{Y})$ simultaneously.

Comparing with problem (19), there are four key differences as follows.

1) The learning tasks are different. This work aims to select a subset of features from the original features, while MPME aims for feature extraction.
2) The definitions of pairwise distances are different. MPME computes distances using all features, while this work computes distances only using the selected features. This can effectively overcome the issue that distance tends to concentrate on high-dimensional data, but MPME cannot. Moreover, the tolerance of distance violation is different so that a directed graph is optimized in this work.
3) Problem (33) is more versatile than MPME. MPME is a special case of this work if $\boldsymbol{\theta}$ consists of all ones, which means all features are equally important. However, this work can learn the importance of features. This is due to the proposed maximum deviation criterion in Section II-A.
4) The optimization algorithms are different. In this paper, we propose a new projection algorithm, which achieves the optimal solution by the bisection search. This will be discussed in Section IV-C.

C. Optimization Algorithm

To solve problem (33), we first reformulate it by the following proposition.

Proposition 4: The partial dual problem of (33) is equivalent to the following optimization problem:

$$\min_{\mathbf{W} \in \mathcal{W}} \min_{\boldsymbol{\theta} \in \boldsymbol{\Theta}} g(\mathbf{W}, \boldsymbol{\theta}) \quad (34)$$

where the objective function is given by

$$g(\mathbf{W}, \boldsymbol{\theta}) = -\frac{1}{2} \sum_{r=1}^{M} \log\det(\gamma \mathbf{I} + 4\theta_r \mathbf{L}_{\mathbf{W}}) + \sum_{i,j} w_{i,j} \psi(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta})$$

with $\mathcal{W} = \{\mathbf{W} \Big| \sum_{j=1}^{N} w_{i,j} = \lambda, w_{i,j} \geq 0, \forall i, j\}$, the its (i, j)th entry of matrix $\mathbf{W}$ is $w_{i,j}$, $\mathbf{L}_{\mathbf{W}} = \mathbf{D} - (\mathbf{W} + \mathbf{W}^T/2)$,

diagonal matrix $\mathbf{D}$ with (i,i)th entry $\sum_j (w_{i,j}+w_{j,i})/2$. The optimal solution of the true density is obtained as $p(\mathbf{Y}) = \prod_{r=1}^{M} p(\mathbf{y}^r)$, with the optimal solution $p(\mathbf{y}^r), \forall r$

$$p(\mathbf{y}^r) \propto \pi_0(\mathbf{y}^r) \exp\left(-\theta_r \sum_{i,j} w_{i,j}(y_{r,i}-y_{r,j})^2\right).$$

Since problems (33) and (19) fall into the same density estimation framework, the properties discussed in Section III-E and the learned structure for the visualization of embeddings in a low-dimensional space hold true for the proposed unsupervised feature selection method.

Projected gradient descent algorithm [49] can be used to solve problem (34). Specifically, problem (34) with respect to $\mathbf{W}$ is equivalent to N Euclidean projection onto simplex problems with respect to each row of $\mathbf{W}$. The work [50] can be applied to solve the projection onto simplex. We then propose a new simple algorithm to find the projection onto Θ in the Section IV-C2.

1) Derivation of Gradients: Let $\mathbf{Q}_r = \gamma \mathbf{I} + 4\theta_r \mathbf{L}_\mathbf{W}$, so $\mathbf{Q}_r^{-1} = \mathbf{U}(\gamma \mathbf{I} + 4\theta_r \mathbf{V})^{-1}\mathbf{U}^T$, where we use the eigendecomposition of $\mathbf{L}_\mathbf{W}$, i.e., $\mathbf{L}_\mathbf{W} = \mathbf{U}\mathbf{V}\mathbf{U}^T$. The partial gradient of $\log\det(\mathbf{Q}_r)$ with respect to $w_{i,j}$ is derived as

$$\partial_{w_{i,j}} \log\det(\mathbf{Q}_r) = 2\theta_r \mathrm{Tr}\left[\mathbf{Q}_r^{-T}\mathbf{A}_{i,j}\right]$$

where matrix $\mathbf{A}_{i,j}$ with the (s,t)th entry defined as

$$\mathbf{A}_{i,j}(s,t) = \begin{cases} -1, & i \neq j \wedge (s=i \wedge j=t \text{ or } s=j \wedge i=t) \\ 1, & s=i=j \text{ or } t=i=j \\ 0, & \text{otherwise.} \end{cases}$$

As a result, the gradient $\partial_{w_{i,j}} g(\mathbf{W}, \boldsymbol{\theta})$ can be written as

$$-\sum_{r=1}^{M} \theta_r \mathrm{Tr}[\mathbf{U}(\gamma \mathbf{I} + 4\theta_r \mathbf{V})^{-1}\mathbf{U}^T \mathbf{A}_{i,j}] + \psi(\mathbf{x}_i, \mathbf{x}_j; \boldsymbol{\theta}) \quad \forall i, j.$$

The partial gradient with respect to $\boldsymbol{\theta}$ is, $\forall r$

$$\partial_{\theta_r} g(\mathbf{W}, \boldsymbol{\theta}) = -2\mathrm{Tr}[\mathbf{U}(\gamma \mathbf{I} + 4\theta_r \mathbf{V})^{-1}\mathbf{U}^T \mathbf{L}_\mathbf{W}] + 2(\mathbf{y}^r)^T \mathbf{L}_\mathbf{W} \mathbf{y}^r.$$

2) Projection Onto Θ: The projection problem with respect to $\boldsymbol{\theta}$ is formulated as

$$\min_{\boldsymbol{\theta}} \frac{1}{2}||\boldsymbol{\theta} - \overline{\mathbf{u}}||^2 : \quad \text{s.t. } \mathbf{1}^T\boldsymbol{\theta} = b, \ 0 \leq \theta_r \leq 1, \ \forall r \tag{35}$$

where $\overline{\mathbf{u}}$ is the intermediate solution of the gradient descent step with respect to $\boldsymbol{\theta}$.

Proposition 5: Problem (35) has a solution

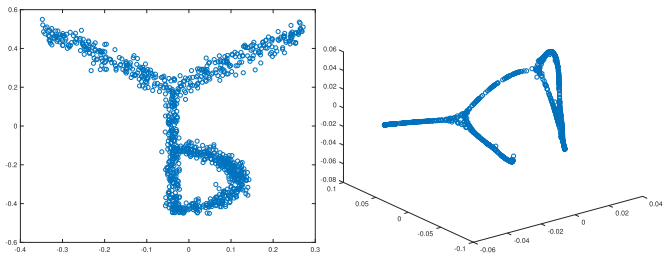
$$\theta_r(\tau) = \begin{cases} 1, & \tau < \overline{u}_r - 1 \\ \overline{u}_r - \tau, & \overline{u}_r - 1 \leq \tau \leq \overline{u}_r \\ 0, & \overline{u}_r < \tau \end{cases} \tag{36}$$

where τ satisfies $\sum_{r=1}^{M} \theta_r(\tau) = b$ and can be obtained by the bisection method.

The pseudocode of the proposed unsupervised feature selection via structure learning (FSL) algorithm is given in Algorithm 2, where the variables $\boldsymbol{\theta}$ and $\mathbf{W}$ are initialized so that no prior information is imposed over the importance of features and the weighted graph.

Algorithm 2 Feature Selection via Structure Learning

1: **Input**: data $\mathbf{X}$, parameters b, λ, γ, and C
2: initialize $\boldsymbol{\theta} = b\mathbf{1}/M$ and $\mathbf{W} = \mathbf{1}\mathbf{1}^T/N$
3: obtain $\boldsymbol{\theta}, \mathbf{W}$ by solving (34) using the projected gradient descent algorithm with the proposed projection algorithm
4: **Output:** $\boldsymbol{\theta}$ and $\mathbf{W}$



(a) (b)

Fig. 3. YB data and its embedded results in 3-D obtained by ESL (Algorithm 1) with perplexity $u = 50$, $\lambda = 0.7$. (a) YB data. (b) $u = 50$, $\lambda = 0.7$ in 3-D.

V. Experiments

Two sets of experiments are conducted to evaluate Algorithms 1 and 2 for ESL and unsupervised feature selection on various synthetic and real-world data sets, respectively.

A. Unsupervised Feature Extraction

1) Parameter Sensitivity Analysis: We investigate the parameter sensitivity of ESL by varying the perplexity u and parameter λ on YB data, a synthetic data of 1000 points in a 2-D space, where the data consist of characters Y and b as shown in Fig. 3(a). For simplicity, we study the influence of both the parameters of ESL with respect to the embedded points and the learned graph structure by varying one and fixing the other. The reduced dimensionality is set to be two. We vary $\lambda \in \{0.6, 0.7, 0.8\}$ and $u \in \{30, 40, 50\}$ and $C = 1$ as the default value due to its least influence on the learned structure and embedded points than the other two parameters.

The visualization results of both embedded points and graph structure learned by ESL are reported in Fig. 4 for varying λ and u. The graph structure visualized by the MATLAB graph toolbox is only used for the illustration of the sparsity of matrix $\mathbf{W}$ and the disconnected components since vertices of the graph are not associated with embedded points. We have the following observations from Fig. 4.

1) The bigger u is, the less sparse the graph matrix $\mathbf{W}$ is. This is consistent with the meaning of u discussed in [8].
2) As λ becomes larger, the sparsity of matrix $\mathbf{W}$ increases, and the learned structure becomes smoother with embedded points of less noise.
3) If u is small and λ approaches to one, the learned graph becomes smoother, where certain detailed structures are lost in 2-D space. However, the correct structure is still maintained in 3-D space as shown in Fig. 3(b).
4) In all combinations of parameters, the embedded points form a nice smooth manifold structure.

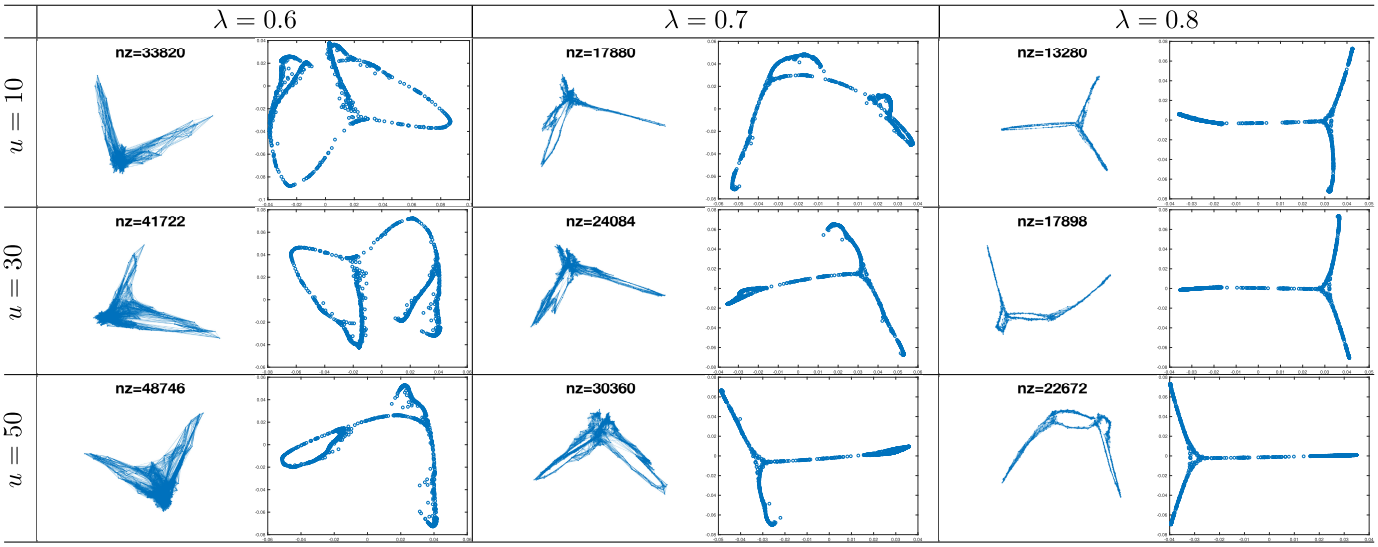


Fig. 4. Parameter sensitivity analysis of ESL on YB data. The graph structure and embedded points obtained by ESL are for $\lambda \in \{0.6, 0.7, 0.8\}$ and perplexity $u \in \{30, 40, 50\}$.

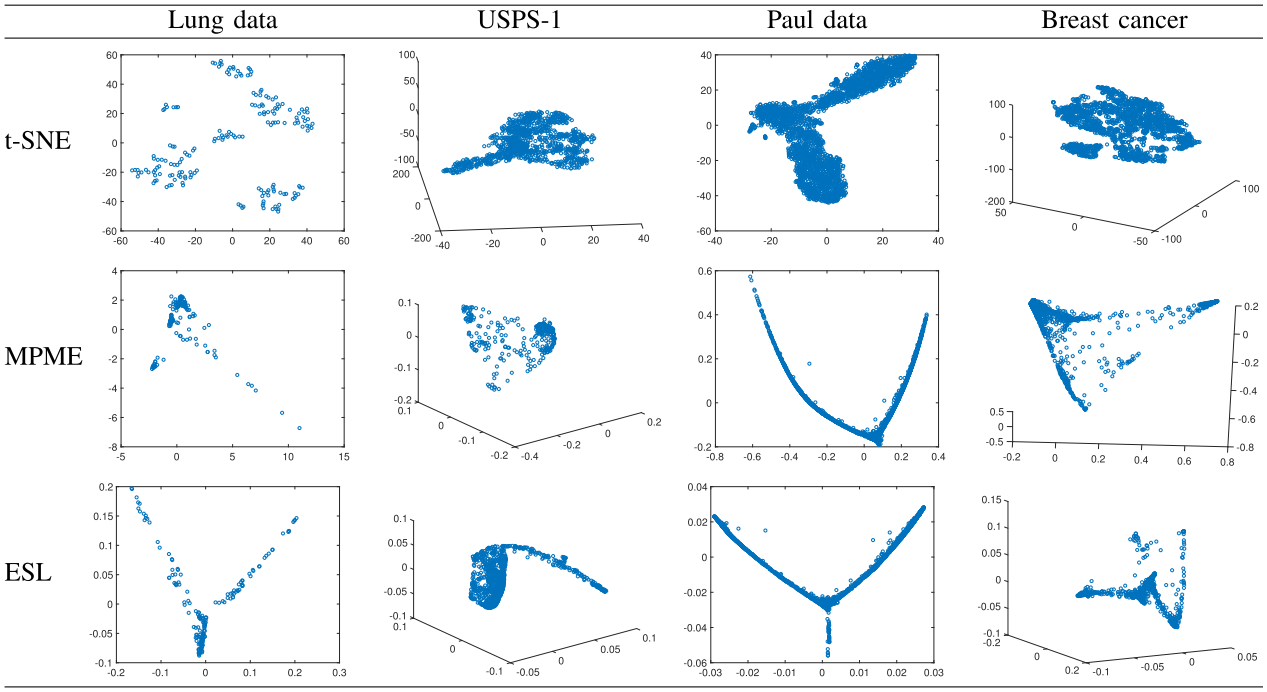


Fig. 5. Embeddings learned by three methods: t-SNE, MPME, and ESL on four real-world data sets.

2) Data Sets With a Single Connected Structure: Four real-world data sets with a single connected structure are evaluated for ESL by comparing with two closely related methods t-SNE [8] and MPME [9]. The Lung data [51] consist of 199 samples with 39 016 genes in total for the lung epithelial cell data analysis. All cells annotated as ciliated cells, Clara cells, or bulk sample from Supplementary Data 5 in [51] are excluded, yielding 183 cells for embedding analysis. The data are processed as described in [52]. The USPS data[1] contain handwritten digits from 0 to 9 with different written styles. Each one is a gray image of size 16×16. USPS-1 is a subset of USPS with only digit 1 and consists of 1100 images. The Paul data are from the Paul experiment [53] and consist of 2730 cells with 3418 genes. The data are processed according to the work [52]. A large-scale, publicly available breast cancer data set [54] contains the expression levels of over 25 000 gene transcripts from 144 normal breast tissue samples and 1989 tumor tissue samples. The data are processed by the same procedure as [9] so that 359 genes are identified for the cancer progression modeling.

Fig. 5 shows the embedded points obtained by the above-mentioned three methods on the four real-world data sets. We tune the perplexity u for both t-SNE and ESL. Except $u = 5$ on Lung data, we find t-SNE on other three data can achieve reasonable results with $u = 30$. We also tune parameter λ in both MPME and ESL. We find that MPME

[1]http://www.cs.nyu.edu/~roweis/data/usps_all.mat

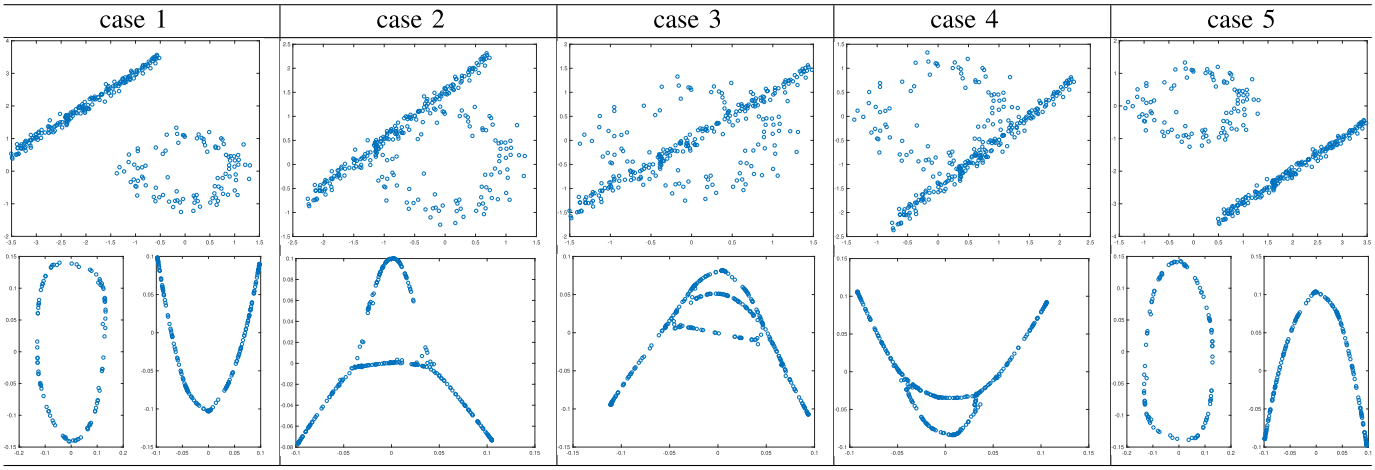


Fig. 6. Embeddings obtained by ESL on data sets, which are generated by varying the locations of points sampled from both a line segment and a circle in the 2-D space. Original 2-D synthetic data (top). Embedded points obtained by ESL (bottom).

prefers a small value such as 10^{-3} and ESL needs a value close to 1. From Fig. 5, we have the following observations. First, t-SNE prefers embeddings of clusterings, while MPME and ESL tend to achieve smooth connected manifold structures. This confirms the motivation of the proposed method in Section III. Second, ESL can achieve much smoother structure of embedded points than MPME and can also recover more detailed structure such as the three branches on Lung data and Paul data. This is consistent with the findings in [52]. Moreover, ESL obtains the cancer progression path that is consistent with the embeddings obtained by MPME but with less noise.

3) Data Sets With Multiple Disconnected Structures: We now exploit the capability of ESL for uncovering multiple disconnected structures from both synthetic and real-world data sets.

To explore the capability of ESL for recognizing multicomponents of the underlying data, we synthesize another data $\mathbf{X} = [x, y] \in \mathbb{R}^{200\times 2}$, namely, Line, by randomly sample points from function $y = (x - 0.5) \times 3.0$, where x is randomly drawn from $[0, 1]$. By simply adding offset values to either x and y, we can obtain new data sets by combining both Line and Circle [55], which consist of various structures as shown in Fig. 6. We run ESL on all five different data sets with $u = 30$ and $\lambda = 0.7$. The embedded points in Fig. 6 successfully capture all the changes of various structures.

We apply ESL to COIL20 data [56]. It contains 20 objects. The images of each object were taken 5° apart as the object is rotated on a turntable and each object has 72 images. The size of each image is 32×32 pixels, with 256 gray levels per pixel. Thus, each image is represented by a 1024-dimensional vector. We run ESL with $u = 10$ and $\lambda = 0.83$. Fig. 7 shows the embedded points with 22 disconnected components, where 20 smooth structures exactly match the 20 objects and the two figures in the third row and second and third columns correspond to images with large noise deviating from its main structure. These results on both synthetic data sets and real-world data imply that ESL can correctly learn embedded points of multiple disconnected structures from noisy data.

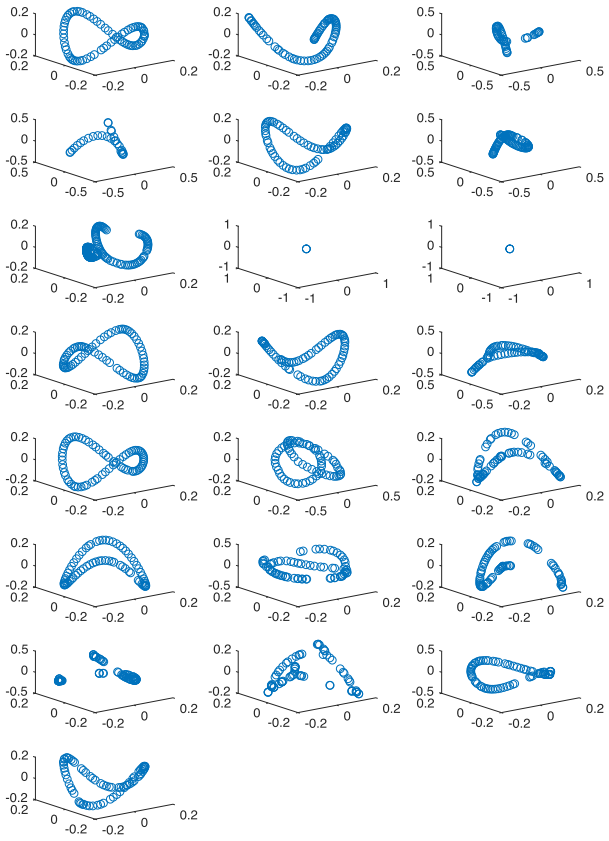


Fig. 7. 22 disconnected structures and embeddings obtained by ESL on COIL20 data with perplexity $u = 10$ and $\lambda = 0.83$.

4) Classification Performance of Embeddings: As shown in Table I, ten data sets taken from the UC Irvine Machine Learning Repository and Statlib repositories are used to evaluate the classification performance of embedded points learned by baseline methods same as those used in the experiments on synthetic data. The reduced dimensionality of data is shown in Table I by preserving 95% of energy of data. Following [11], we use the leave-one-out cross-validation accuracy as the

TABLE I

LEAVE-ONE-OUT CROSS VALIDATION ACCURACY OF ONE-NN CLASSIFIER OVER TEN DATA SETS. N IS THE NUMBER OF DATA POINTS. c IS THE TRUE NUMBER OF CLUSTERS. D IS THE ORIGINAL DIMENSIONALITY, AND d IS THE REDUCED DIMENSIONALITY. THE BEST RESULTS ARE IN BOLD

	Iris	CMU-PIE	COIL20	Isolet	Pendigits	Satimage	USPS	Vehicle	Segment	Letter
(N, c)	(150, 3)	(3329, 68)	(1440, 20)	(3119, 2)	(3498, 10)	(4435, 6)	(2007, 10)	(846, 4)	(231, 7)	(5000, 26)
(D, d)	(4, 2)	(1024, 39)	(2014, 84)	(617, 165)	(16, 9)	(36, 6)	(256, 32)	(18, 6)	(19, 7)	(16, 12)
LLE [19]	0.9467	0.9655	0.9965	0.9298	0.9760	0.8570	0.9013	0.6537	0.9623	0.8960
LE [6]	0.9133	0.6248	0.9833	0.9368	0.9714	0.8586	0.9023	0.5981	0.9398	0.7436
MVU [11]	0.6533	0.4662	0.7660	0.8035	0.9737	0.8607	0.7693	0.5579	0.9342	0.6042
KPCA[18]	0.9000	0.2701	0.5583	0.7086	0.9883	0.8462	0.3303	0.5615	0.9550	0.8488
GPLVM [22]	0.9333	0.9787	**1.0000**	0.8410	0.9866	0.8884	0.5944	0.5898	**0.9688**	0.8974
MEU [24]	0.8867	0.9507	**1.0000**	0.9349	0.9840	0.8652	0.9312	0.6407	0.9537	0.1244
SMCE [25]	0.9400	0.9612	**1.0000**	0.9314	0.9806	0.8848	0.9307	0.6832	0.9398	0.8790
t-SNE [8]	**0.9600**	0.9751	**1.0000**	0.9468	0.9909	**0.9037**	0.9292	0.6738	0.9671	**0.9134**
MPME [9]	0.9467	0.9588	**1.0000**	0.9357	0.9900	0.8656	**0.9322**	0.6525	0.9558	0.8520
ESL(ours)	**0.9600**	**0.9796**	**1.0000**	**0.9481**	**0.9917**	0.8875	0.9292	**0.6927**	0.9662	0.9070

criterion for evaluating one-NN classifier on the embeddings learned by these baseline methods. For methods that require K-NN graph as input, we tune $K \in \{5, 10, 15, 20, 30, 50\}$. We tune the parameter $\lambda \in \{0.01, 0.1, 1, 10\}$ for SMCE. Other parameters are set as the default values in the drtoolbox[2]. In addition, we tune $\lambda \in [0.1, 10]$ for MPME, $u \in \{20, 30, 40, 50\}$ and $\lambda \in [0.1, 0.9]$ for ESL. The best results are reported for every baseline method by tuning their own parameters.

Table I shows the leave-one-out cross-validation accuracy of one-NN classifier over the embeddings learned by ten methods on ten benchmark data sets. It is clear to see that ESL is competitive to t-SNE in terms of classification accuracy and produces much better results than the others including MPME. As shown in [8], t-SNE helps to achieve good classification performance by learning a new embedding of original data. The learning criterion of t-SNE is more suitable for clustering/classification, but less appropriate for learning skeleton structures in a latent space as observed in Fig. 5. These results imply that ESL is not only better than MPME for learning skeleton structures in latent spaces from high-dimensional data but also can achieve competitive classification performance on the learned embedded points when compared with t-SNE.

B. Unsupervised Feature Selection

We conduct various experiments, including parameter sensitivity analysis of FSL, quantitative evaluation in terms of clustering performance compared with baselines, and structure learned by FSL.

1) Experiment Setting: To evaluate our FSL method, we compare it with nine state-of-the-art unsupervised feature selection methods on various types of high-dimensional data sets. Table II shows the statistics of the data sets used in the experiments. Following the widely used setting [5], the number of features is selected and the performance is then evaluated for each compared method over a grid of fixed numbers of features. The number of features is searched over the grid $b \in \{20, 40, \ldots, 300\}$.

We partition various existing unsupervised feature selection methods into groups based on their required assumptions and compare the representatives from each group.

[2]https://lvdmaaten.github.io/drtoolbox/

TABLE II

STATISTICS OF THE DATA SETS USED FOR UNSUPERVISED FEATURE SELECTION

datasets	N	D	#classes	selected features
Yale	165	1,024	15	$\{20, 40, \ldots, 300\}$
COIL20	1,440	1,024	20	$\{20, 40, \ldots, 300\}$
warpAR10P	130	2,400	10	$\{20, 40, \ldots, 300\}$
TOX	171	5,748	4	$\{20, 40, \ldots, 300\}$
two-moon	400	1,002	N/A	2
teapot	400	43,028	N/A	N/A

These representatives include: 1) manifold assumption: LS [7] and SPEC [28]; 2) manifold + clustering: MCFS [29], NDFS [30], and RUFS [31]; 3) similarity preservation: SPFS [33]; and 4) structure learning, LLCFS [38], FSASL [5], and SOGFS [39]. For fair comparisons, we fix the size of neighborhoods as 5 by following [5], [29]. For methods based on Gaussian kernels, we search the kernel width in the grid $\{2^{-3}, 2^{-2}, \ldots, 2^3\}\delta_0$, where δ_0 is the mean distance between any two data points. The hyperparameters are tuned in the grid $\{10^{-3}, 10^{-2}, \ldots, 10^3\}$. For hyperparameters of some methods that do not fall into this grid, we adopt the default searching grid reported in their studies. According to these grids, we tune these parameters for all methods by using the grid search strategy [31]. For methods that do not directly select the given number of features, their parameters are tuned so that the number of selected features falls into the same grid.

2) Parameter Sensitivity Analysis: According to Section IV-B, the proposed formulation (34) has three parameters b, λ, and γ. As discussed before, large value γ promotes the sparsity of $\boldsymbol{\theta}$. In the experiments, we set $\gamma = 10^2$, and study two parameters λ and b by fixing one and varying the other. We investigate the changes by varying each parameter according to three criteria: clustering performance including accuracy and normalized mutual information (NMI) [29], the number of nonzero entries of $\boldsymbol{\theta}$ for the selected features, and the proportion of zero entries in $\mathbf{W}$ for measuring the sparsity of the learned graph.

Fig. 8 shows the results for the parameter sensitivity analysis of the proposed FSL method on Yale by fixing one parameter and varying the other. We have the following observations. According to Fig. 8(a)–(c), the increase of b leads to the increase of the selected features and an ascending trend of the sparsity of the learned graph. In contrast, the increase of λ

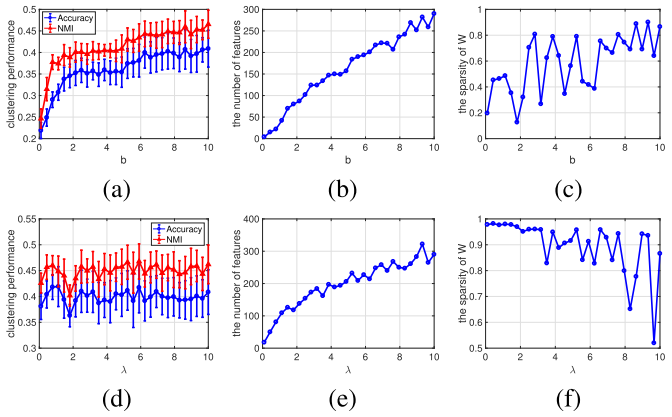


Fig. 8. Parameter sensitivity analysis of FSL on Yale. Subplots for fixing $\lambda = 10$ and varying b are (a) clustering performance in terms of accuracy and NMI with error bars, (b) number of selected features, and (c) sparsity of $\mathbf{W}$. (d)–(f) corresponding results by fixing $b = 10$ and varying λ.

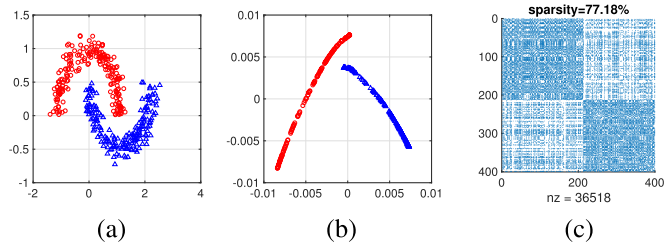


Fig. 9. Graph structure and embeddings obtained by FSL on the synthetic two-moon data for unsupervised feature selection with parameters $\gamma = 100$, $\lambda = 10$, and $b = 1$. (a) Original data. (b) Embeddings. (c) Adjacency matrix.

leads to the increase of the selected features but a descending trend of the sparsity of the graph as shown in Fig. 8(d)–(f). This difference is useful since the number of features and the sparsity of the graph are not necessarily correlated (i.e., data dependent), but they affect each other. In other words, we can tune λ and b properly to get the desired number of features and sparsities of the learned graph. On the other hand, we find that a good $\mathbf{W}$ is able to contribute positively to the clustering performance for a wide range of selected features as shown in Fig. 8(d). These observations are consistent with the motivations of the proposed method for simultaneously selecting features and learning graph structures from data.

3) Structures Learned by FSL: We investigate the structures learned by FSL over the embedded points and the set of selected features in detail by performing experiments on two data sets with known structures. The two-moon data is a 2-D data for smooth structure learning [9]. We synthesize an augmented data from the original two-moon data by adding noisy data with 1000 dimensions sampled independent identically distributed (i.i.d.), from a uniform distribution in $[0, 1]$. The goal is to find the two separate curves by selecting the top two features. The teapot image data consists of 400 RGB images [57]. These images were taken successively as a teapot was rotated 360°. Each image consists of 76×101 pixels and is represented as a vector. The ideal graph structure is a circle on which all images are well organized.

Experimental results on two data sets, two-moon and teapot, are shown in Figs. 9 and 10, respectively. The graph structures learned by the proposed method can be shown in two different

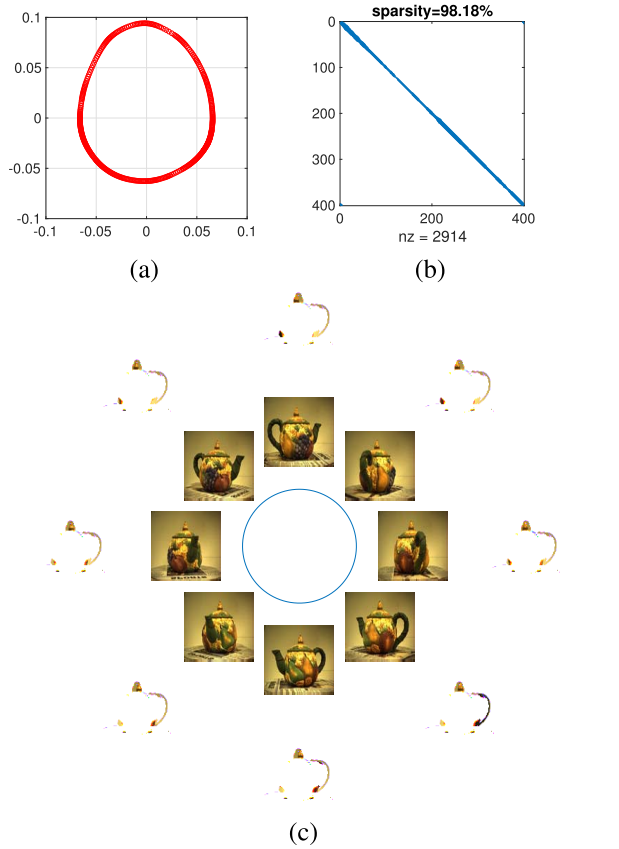


Fig. 10. Graph structure and embeddings obtained by FSL on the teapot data for unsupervised feature selection with parameters $\gamma = 100$, $\lambda = 0.1$, and $b = 0.7$. Best visualized in color. (a) Embeddings. (b) Adjacency matrix. (c) Images and their masked counterparts.

perspectives: the embedded points in the 2-D space and the adjacency matrix. For the two-moon data, our method can successfully find the top two features with $\theta_1 = 0.5758$ and $\theta_2 = 0.4242$, and other entries of $\boldsymbol{\theta}$ are 0s. From Fig. 9, we can see that the embedded points [Fig. 9(b)] form exactly two separate curves for the moon shapes and the adjacency matrix [Fig. 9(c)] shows more connections within each moon than between two moons. On the teapot data, 1074 features are selected from 43 028 features. Fig. 10(a) shows the circular structure formed by the embedded points in 2-D space. The adjacency matrix in Fig. 10(b) demonstrates a thin diagonal structure, which means that the connectivities are only presented when two images are close enough on the circular structure. Moreover, the mask formed by the selected features is applied to each image so as to investigate the importance of these features. By carefully looking at these masked images, we can find that the pixels selected vary smoothly according to the circular structure of the data. These observations imply that our FSL method not only is able to select a set of features to represent the underlying structure of the data but also provides a natural embedding solution.

4) Clustering With Selected Features: We evaluate the performance of compared methods in terms of the k-means clustering by two widely used metrics: accuracy and NMI. The number of clusters of the k-means clustering method is set to be the number of true clusters. For our method, we tune $b \in [0.1, 40]$ and $\lambda \in [0.1, 10]$. Since the results of

TABLE III

CLUSTERING RESULTS MEASURED BY ACCURACY AND NMI (MEAN ± STD IN PERCENT) REPORTED FOR TEN METHODS ON FOUR DATA SETS. THE BEST RESULTS ARE IN BOLD

Methods	Accuracy				NMI			
	Yale	COIL20	warpAR10P	TOX_171	Yale	COIL20	warpAR10P	TOX_171
LS[7]	43.00 ± 4.58	57.58 ± 4.08	31.15 ± 4.46	42.19 ± 2.24	48.44 ± 2.91	71.00 ± 1.51	32.67 ± 2.26	12.30 ± 1.88
SPEC[28]	44.03 ± 3.27	54.43 ± 3.11	42.42 ± 1.75	43.45 ± 3.48	49.67 ± 2.77	67.89 ± 1.58	44.25 ± 1.50	11.99 ± 2.71
MCFS[29]	39.24 ± 3.49	59.02 ± 5.14	26.38 ± 3.93	44.33 ± 3.81	45.41 ± 2.50	72.54 ± 2.08	23.31 ± 4.17	15.56 ± 3.26
NDFS[30]	39.15 ± 2.75	**65.25 ± 4.29**	24.31 ± 3.54	48.01 ± 1.20	45.36 ± 2.37	75.49 ± 2.28	20.83 ± 5.60	22.92 ± 1.86
RUFS[31]	39.55 ± 4.40	63.14 ± 4.54	33.50 ± 2.92	**50.61 ± 0.58**	45.35 ± 3.02	75.35 ± 1.69	28.57 ± 3.93	**28.12 ± 2.81**
SPFS[33]	42.30 ± 4.48	62.48 ± 4.74	38.19 ± 3.40	42.08 ± 2.28	49.66 ± 2.48	75.43 ± 2.20	40.22 ± 3.83	12.96 ± 3.65
LLCFS[38]	37.33 ± 2.24	57.85 ± 4.88	43.65 ± 5.23	41.78 ± 2.61	42.78 ± 1.90	71.93 ± 1.96	41.66 ± 4.28	12.93 ± 3.07
FSASL[5]	38.45 ± 3.02	62.83 ± 3.98	29.38 ± 2.48	50.91 ± 1.27	44.57 ± 1.98	**75.66 ± 2.20**	25.25 ± 2.67	26.82 ± 2.49
SOGFS[39]	40.97 ± 2.87	56.60 ± 3.41	42.96 ± 3.29	44.53 ± 2.71	47.43 ± 2.94	71.01 ± 1.71	45.76 ± 3.02	20.46 ± 3.70
FSL (ours)	**45.76 ± 3.78**	63.65 ± 5.03	**43.92 ± 3.74**	46.17 ± 4.03	**51.05 ± 2.78**	75.57 ± 1.85	**48.50 ± 2.79**	28.05 ± 1.48

the k-means clustering method depend on the initialization, we perform clustering with 20 random initializations and report the mean and standard deviation.

The clustering results in terms of accuracy and NMI are shown in Table III by comparing ten methods on four data sets through fine-tuning parameters of each method using the greedy search strategy. For each compared method, we also report the mean and its standard deviation for performance evaluation. From Table III, we have the following observations. Our FSL method does not depend on either the prefixed manifold or the clustering assumption, and its performance is competitive to or better than the existing unsupervised feature selection methods. SPFS with similarity preserving is worse than our method with distance preservation on the four data sets. Our method performs competitively to methods such as NDFS, RUFS, and FSASL in terms of both accuracy and NMI. These observations demonstrate that our FSL method is effective for unsupervised feature selection.

VI. CONCLUSION

In this paper, we proposed a density estimation approach to tackle unsupervised dimensionality reduction problem with a unified framework for both feature extraction and feature selection. Our approach not only obtains the proper embeddings in a low-dimensional space for feature extraction and selects a proper set of features from original high-dimensional data for feature selection but also learns a similarity matrix or graph structure of the true data drawn from the learned density function. Distinguishing from various existing methods, our learning criteria are very different due to the unique assumption on distance preservation and structure learning. Two novel methods were proposed based on the framework. Extensive experiments demonstrate that our proposed methods can achieve competitive quantitative results in terms of discriminant evaluation performance and are able to obtain the embeddings of smooth skeleton structures and select optimal features to unveil the correct graph structures of high-dimensional data sets. The models of this work do not explore the property of matrices for computational consideration and the multimodality data [58], [59]. In the future work, we will leverage some advanced matrix analysis algorithms [60], [61] to reduce the computational complexity for large-scale data and extend our models for multimodality problem.

REFERENCES

[1] S. Khalid, T. Khalil, and S. Nasreen, "A survey of feature selection and feature extraction techniques in machine learning," in *Proc. IEEE Sci. Inf. Conf. (SAI)*, Aug. 2014, pp. 372–378.

[2] I. Guyon and A. Elisseeff, "An introduction to variable and feature selection," *J. Mach. Learn. Res.*, vol. 3, pp. 1157–1182, Jan. 2003.

[3] W. M. Hartmann, "Dimension reduction vs. variable selection," in *Int. Workshop Appl. Parallel Comput.* Berlin, Germany: Springer, 2004, pp. 931–938.

[4] J. G. Dy and C. E. Brodley, "Feature selection for unsupervised learning," *J. Mach. Learn. Res.*, vol. 5, pp. 845–889, Jan. 2004.

[5] L. Du and Y.-D. Shen, "Unsupervised feature selection with adaptive structure learning," in *Proc. ACM SIGKDD*, 2015, pp. 209–218.

[6] M. Belkin and P. Niyogi, "Laplacian eigenmaps and spectral techniques for embedding and clustering," in *Proc. NIPS*, vol. 14, 2001, pp. 585–591.

[7] X. He, D. Cai, and P. Niyogi, "Laplacian score for feature selection," in *Proc. NIPS*, 2006, pp. 507–514.

[8] L. van der Maaten and G. Hinton, "Visualizing data using t-SNE," *J. Mach. Learn. Res.*, vol. 9, no. 2579–2605, p. 85, 2008.

[9] Q. Mao, L. Wang, and I. W. Tsang, "A unified probabilistic framework for robust manifold learning and embedding," *Mach. Learn.*, vol. 106, no. 5, pp. 627–650, 2017.

[10] L. Wang, Q. Mao, and I. W. Tsang, "Latent smooth skeleton embedding," in *Proc. AAAI*, 2017, pp. 2703–2709.

[11] K. Q. Weinberger, F. Sha, and L. K. Saul, "Learning a kernel matrix for nonlinear dimensionality reduction," in *Proc. ICML*, 2004, p. 106.

[12] K. Q. Weinberger, B. D. Packer, and L. K. Saul, "Nonlinear dimensionality reduction by semidefinite programming and kernel matrix factorization," in *Proc. AISTATS*, 2005, pp. 381–388.

[13] C. Hou, F. Nie, D. Yi, and Y. Wu, "Feature selection via joint embedding learning and sparse regression," in *Proc. IJCAI*, vol. 22, no. 1, 2011, p. 1324.

[14] Z. Li, J. Liu, Y. Yang, X. Zhou, and H. Lu, "Clustering-guided sparse structural learning for unsupervised feature selection," *IEEE Trans. Knowl. Data Eng.*, vol. 26, no. 9, pp. 2138–2150, Sep. 2014.

[15] L. van der Maaten, E. O. Postma, and H. J. van den Herik, "Dimensionality reduction: A comparative review," Tilburg Centre Creative Comput., Tilburg Univ., Tilburg, The Netherlands, Tech. Rep. TiCC-TR 2009-005, 2009. [Online]. Available: https://lvdmaaten.github.io/publications/papers/TR_Dimensionality_Reduction_Review_2009.pdf

[16] C. J. Burges, "Dimension reduction: A guided tour," *Found. Trends Mach. Learn.*, vol. 2, no. 4, pp. 275–365, 2010.

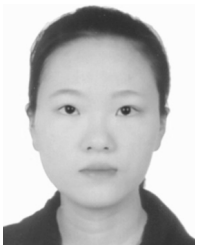
[17] J. T. Jolliffe, *Principal Component Analysis*. Berlin, Germany: Springer-Verlag, 1986.

[18] B. Schölkopf, A. Smola, and K. Müller, "Kernel principal component analysis," in *Advances in Kernel Methods: Support Vector Learning*. Cambridge, MA, USA: MIT Press, 1999, pp. 327–352. [Online]. Available: https://dl.acm.org/citation.cfm?id=299113

[19] L. K. Saul and S. T. Roweis, "Think globally, fit locally: Unsupervised learning of low dimensional manifolds," *J. Mach. Learn. Res.*, vol. 4, pp. 119–155, Dec. 2003.

[20] Z. Zhang and H. Zha, "Principal manifolds and nonlinear dimensionality reduction via tangent space alignment," *J. Shanghai Univ.*, vol. 8, no. 4, pp. 406–424, 2004.

[21] G. Hinton and S. Roweis, "Stochastic neighbor embedding," in *Proc. NIPS*, 2003, pp. 857–864.
[22] N. Lawrence, "Probabilistic non-linear principal component analysis with Gaussian process latent variable models," *J. Mach. Learn. Res.*, vol. 6, pp. 1783–1816, Nov. 2005.
[23] M. E. Tipping and C. M. Bishop, "Probabilistic principal component analysis," *J. Roy. Statist. Soc. B, Stat. Methodol.*, vol. 61, no. 3, pp. 611–622, 1999.
[24] N. D. Lawrence, "A unifying probabilistic perspective for spectral dimensionality reduction: Insights and new models," *J. Mach. Learn. Res.*, vol. 13, no. 1, pp. 1609–1638, Jan. 2012.
[25] E. Elhamifar and R. Vidal, "Sparse manifold clustering and embedding," in *Proc. NIPS*, 2011, pp. 55–63.
[26] B. Cheng, J. Yang, S. Yan, Y. Fu, and T. S. Huang, "Learning with ℓ^1-graph for image analysis," *IEEE Trans. Image Process.*, vol. 19, no. 4, pp. 858–866, Apr. 2010.
[27] B. T. Lake and J. B. Tenenbaum, "Discovering structure by learning sparse graph," in *Proc. 33rd Annu. Cognit. Sci. Conf.*, 2010, pp. 778–784.
[28] Z. Zhao and H. Liu, "Spectral feature selection for supervised and unsupervised learning," in *Proc. ACM ICML*, 2007, pp. 1151–1157.
[29] D. Cai, C. Zhang, and X. He, "Unsupervised feature selection for multi-cluster data," in *Proc. ACM SIGKDD*, 2010, pp. 333–342.
[30] Z. Li, Y. Yang, J. Liu, X. Zhou, and H. Lu, "Unsupervised feature selection using nonnegative spectral analysis," in *Proc. AAAI*, vol. 2012, 2012, pp. 1026–1032.
[31] M. Qian and C. Zhai, "Robust unsupervised feature selection," in *Proc. IJCAI*, 2013, pp. 1621–1627.
[32] L. Shi, L. Du, and Y.-D. Shen, "Robust spectral learning for unsupervised feature selection," in *Proc. ICDM*, 2014, pp. 977–982.
[33] Z. Zhao, L. Wang, H. Liu, and J. Ye, "On similarity preserving feature selection," *IEEE Trans. Knowl. Data Eng.*, vol. 25, no. 3, pp. 619–632, Mar. 2013.
[34] X. Liu, L. Wang, J. Zhang, J. Yin, and H. Liu, "Global and local structure preservation for feature selection," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 6, pp. 1083–1095, Jun. 2013.
[35] X. Wei, B. Cao, and P. S. Yu, "Nonlinear joint unsupervised feature selection," in *Proc. ICDM*. Philadelphia, PA, USA: SIAM, 2016, pp. 414–422.
[36] L. Song, A. Smola, A. Gretton, K. M. Borgwardt, and J. Bedo, "Supervised feature selection via dependence estimation," in *Proc. ACM ICML*, 2007, pp. 823–830.
[37] X. Wei and P. S. Yu, "Unsupervised feature selection by preserving stochastic neighbors," in *Proc. Mach. Learn. Res. (PMLR)*, Cadiz, Spain, 2016, pp. 995–1003. [Online]. Available: http://proceedings.mlr.press/v51/wei16.html
[38] H. Zeng and Y.-M. Cheung, "Feature selection and kernel learning for local learning-based clustering," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 33, no. 8, pp. 1532–1547, Aug. 2011.
[39] F. Nie, W. Zhu, and X. Li, "Unsupervised feature selection with structured graph optimization," in *Proc. AAAI*, 2016, pp. 1302–1308.
[40] S. Boyd and L. Vandenberghe, *Convex Optimization*. Cambridge, U.K.: Cambridge Univ. Press, 2004.
[41] R. Krishnapuram and J. M. Keller, "A possibilistic approach to clustering," *IEEE Trans. Fuzzy Syst.*, vol. 1, no. 2, pp. 98–110, May 1993.
[42] R. Krishnapuram and J. M. Keller, "The possibilistic C-means algorithm: Insights and recommendations," *IEEE Trans. Fuzzy Syst.*, vol. 4, no. 3, pp. 385–393, Aug. 1996.
[43] R. H. Byrd, P. Lu, J. Nocedal, and C. Zhu, "A limited memory algorithm for bound constrained optimization," *SIAM J. Sci. Comput.*, vol. 16, no. 5, pp. 1190–1208, 1995.
[44] U. von Luxburg, "A tutorial on spectral clustering," *Statist. Comput.*, vol. 17, no. 4, pp. 395–416, 2007.
[45] L. Xiao, J. Sun, and S. Boyd, "A duality view of spectral methods for dimensionality reduction," in *Proc. ICML*, 2006, pp. 1041–1048.
[46] M. Tan, L. Wang, and I. W. Tsang, "Learning sparse SVM for feature selection on very high dimensional datasets," in *Proc. ICML*, 2010, pp. 1047–1054.
[47] A. L. Berger, V. J. D. Pietra, and S. A. D. Pietra, "A maximum entropy approach to natural language processing," *Comput. Linguistics*, vol. 22, no. 1, pp. 39–71, 1996.
[48] J. Zhu, N. Chen, and E. P. Xing, "Bayesian inference with posterior regularization and applications to infinite latent SVMs," *J. Mach. Learn. Res.*, vol. 15, pp. 1799–1847, Jan. 2014.
[49] M. Schmidt, E. van den Berg, M. P. Friedlander, and K. Murphy, "Optimizing costly functions with simple constraints: A limited-memory projected quasi-Newton algorithm," in *Proc. AISTATS*, 2009, pp. 456–463.
[50] J. Duchi, S. Shalev-Shwartz, Y. Singer, and T. Chandra, "Efficient projections onto the ℓ_1-ball for learning in high dimensions," in *Proc. ACM ICML*, 2008, pp. 272–279.
[51] B. Treutlein *et al.*, "Reconstructing lineage hierarchies of the distal lung epithelium using single-cell RNA-seq," *Nature*, vol. 509, no. 7500, pp. 371–375, 2014.
[52] X. Qiu *et al.*, "Reversed graph embedding resolves complex single-cell trajectories," *Nature Methods*, vol. 14, no. 10, pp. 979–982, 2017.
[53] F. Paul *et al.*, "Transcriptional heterogeneity and lineage commitment in myeloid progenitors," *Cell*, vol. 163, no. 7, pp. 1663–1677, 2015.
[54] C. Curtis *et al.*, "The genomic and transcriptomic architecture of 2,000 breast tumours reveals novel subgroups," *Nature*, vol. 486, pp. 346–352, Jun. 2012.
[55] B. Kégl, A. Krzyzak, T. Linder, and K. Zeger, "Learning and design of principal curves," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 22, no. 3, pp. 281–297, Mar. 2000.
[56] S. A. Nene, S. K. Nayar, and H. Murase, "Columbia object image library (coil-20)," Dept. Comput. Sci., Columbia Univ., New York, NY, USA, Tech. Rep. CUCS-005-96, 1996. [Online]. Available: http://www1.cs.columbia.edu/CAVE/publications/pdfs/Nene_TR96.pdf
[57] K. Q. Weinberger and L. K. Saul, "An introduction to nonlinear dimensionality reduction by maximum variance unfolding," in *Proc. AAAI*, vol. 6, 2006, pp. 1683–1686.
[58] S. Bai, X. Bai, L. J. Latecki, and Q. Tian, "Multidimensional scaling on multiple input distance matrices," in *Proc. AAAI*, 2017, pp. 1281–1287.
[59] F. Nie, G. Cai, J. Li, and X. Li, "Auto-weighted multi-view learning for image clustering and semi-supervised classification," *IEEE Trans. Image Process.*, vol. 27, no. 3, pp. 1501–1511, Mar. 2018.
[60] C. Boutsidis, P. Drineas, P. Kambadur, E.-M. Kontopoulou, and A. Zouzias, "A randomized algorithm for approximating the log determinant of a symmetric positive definite matrix," *Linear Algebra Appl.*, vol. 533, pp. 95–117, Nov. 2017.
[61] I. Han, D. Malioutov, and J. Shin, "Large-scale log-determinant computation through stochastic chebyshev expansions," in *Proc. Int. Conf. Mach. Learn.*, 2015, pp. 908–917.



Li Wang received the bachelor's degree in information and computing science from the China University of Mining and Technology, Jiangsu, China, in 2006, the master's degree in computational mathematics from Xi'an Jiaotong University, Shaanxi, China, in 2009, and the Ph.D. degree from Department of Mathematics, University of California at San Diego, San Diego, CA, USA, in 2014.

She was as the Post-Doctoral Fellow with Brown University, Providence, RI, USA, in 2014, and the University of Victoria, Victoria, BC, Canada, in 2015. From 2015 to 2017, she was a Research Assistant Professor with the Department of Mathematics, Statistics, and Computer Science, University of Illinois at Chicago, Chicago, IL, USA. She is currently an Assistant Professor with the Department of Mathematics, University of Texas at Arlington, Arlington, TX, USA. Her current research interests include large-scale optimization, polynomial optimization, and machine learning.

Ren-cang Li received the B.S. degree in computational mathematics from Xiamen University, Xiamen, China, in 1985, the M.S. degree in computational mathematics from the Chinese Academy of Science, Beijing, China, in 1988, and the Ph.D. degree in applied mathematics from the University of California at Berkeley, Berkeley, CA, USA, in 1995.

He is currently a Professor with the Department of Mathematics, University of Texas at Arlington, Arlington, TX, USA. He received the 1995 Householder Fellowship in Scientific Computing by the Oak Ridge National Laboratory, a Friedman Memorial Prize in applied mathematics from the University of California at Berkeley in 1996, and a CAREER Award from the U.S. National Science Foundation in 1999. His current research interest includes floating-point support for scientific computing, large and sparse linear systems, eigenvalue problems, model reduction, machine learning, and unconventional schemes for differential equations.