This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

K-Deep Simplex: Manifold Learning via Local Dictionaries

Abiy Tasissa, Pranay Tankala, James M. Murphy, and Demba Ba

***Abstract*—We propose K-*Deep Simplex (KDS)* which, given a set of data points, learns a dictionary comprising synthetic landmarks, along with representation coefficients supported on a simplex. KDS employs a local weighted ℓ_1 penalty that encourages each data point to represent itself as a convex combination of nearby landmarks. We solve the proposed optimization program using alternating minimization and design an efficient, interpretable autoencoder using algorithm unrolling. We theoretically analyze the proposed program by relating the weighted ℓ_1 penalty in KDS to a weighted ℓ_0 program. Assuming that the data are generated from a Delaunay triangulation, we prove the equivalence of the weighted ℓ_1 and weighted ℓ_0 programs. We further show the stability of the representation coefficients under mild geometrical assumptions. If the representation coefficients are fixed, we prove that the sub-problem of minimizing over the dictionary yields a unique solution. Further, we show that low-dimensional representations can be efficiently obtained from the covariance of the coefficient matrix. Experiments show that the algorithm is highly efficient and performs competitively on synthetic and real data sets.**

***Index Terms*—Manifold learning, dictionary learning, clustering, structured deep learning**

I. Introduction

Consider observations of the form $(\mathbf{x}_i, \mathbf{y}_i)_{i=1}^n$ with $\mathbf{x}_i \in \mathcal{R}^m$ and $\mathbf{y}_i \in \mathcal{R}^d$ denoting predictor and response variables respectively. We assume that $\mathbf{y}_i = f(\mathbf{x}_i) + \varepsilon_i$ where ε_i represents random i.i.d noise. A ubiquitous model is the standard linear regression which first posits that f is linear and correspondingly estimates the model parameters via different methods (e.g., least squares). Rather than fixing a parametric model as in linear regression, non-parametric models learn the relation f from the data with minimal assumption on f (e.g., smoothness). One popular class of non-parametric models is the local linear regression model [1]–[4]. In contrast to linear regression which assumes a global form of f, local regression is based on approximating f locally using linear functions. To be precise, the local linear fit at a point $\mathbf{x}_i$ is defined using a weight function w_i that depends on distances to all other training data points (i.e., less weight is assigned to points far from $\mathbf{x}_i$). Unlike the linear regression model where the global linear function is only needed for prediction at a test point, the locally linear model depends on the adaptive weight function and hence is non-parametric. One downside of this model is the curse of dimensionality where locality defined via distance functions implies that the weight function either considers nearly no neighbors or nearly all neighbors. To circumvent this limitation, dimensionality reduction-based approaches have been studied [5], [6]. Recent works have also explored local regression with new regularizations and recast it as an optimization problem over a suitably defined graph [7]–[9].

In this paper, we consider the unsupervised learning problem where we only have access to high-dimensional data $(\mathbf{y}_i)_{i=1}^n$ with $\mathbf{y}_i \in \mathcal{R}^d$. This setting arises in many applications and the raw high-dimensional representation presents challenges for computation, visualization, and analysis. The *manifold hypothesis* posits that many high-dimensional datasets can be approximated by a low-dimensional manifold or mixture thereof. Hereafter, a k-dimensional submanifold $\mathcal{M}$ is a subset of $\mathcal{R}^d$ which locally is a flat k-dimensional Euclidean space [10]. If the data lie on or near a *linear* subspace, principal component analysis (PCA) can be used to obtain a low-dimensional representation. But, PCA may fail to preserve nonlinear structures. Nonlinear dimensionality reduction techniques [11]–[15] obtain low-dimensional representations while preserving local geometric structures of the data.

Our main motivation is to develop a model akin to local linear regression in the unsupervised setting. In fact, one of the critical parts of the local regression model is determining the neighborhood radius for each point such that the linear approximation is applied within the specified radius. We note that if the radius is set “large”, the linear approximation is sub-optimal. On the other hand, if the radius is set “small”, the locally linear estimate will be poor as it will only consider very few points. Given these extremes, determining the neighborhood radius, referred as the bandwidth function in the local regression literature [1], is of fundamental importance. A similar challenge also occurs in manifold learning algorithms in determining the number of neighbors (e.g., in locally linear embedding (LLE) [13]).

Herein, to build our model for the unsupervised setting, we use synthetic points for the locally linear approximation. To be precise, rather than considering the whole data set and considering neighboring points, we build local approximations by employing synthetic points that are to be learned. This approach resembles archetypal analysis [16], [17] where data points $(\mathbf{y}_i)_{i=1}^n$ with $\mathbf{y}_i \in \mathcal{R}^d$ are expressed as a convex combination of points $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m$ i.e., $\mathbf{y}_i = \sum_{j=1}^m x_j \mathbf{a}_j$ where $x_j \geqslant 0\,\forall j$ and $\sum_{j=1}^m x_j = 1$. The set of points $\{\mathbf{a}_i\}_{i=1}^m$ are known as the archetypes. In the original archetypal analysis paper [16], an alternating least squares problem is proposed to

Abiy Tasissa and James M. Murphy are in the Department of Mathematics, Tufts University.

Pranay Tankala and Demba Ba are at the school of Engineering and Applied Sciences, Harvard University.

This paper has supplementary downloadable material available at http://ieeexplore.ieee.org., provided by the authors. The material includes additional theoretical analysis and details of numerical experiments.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

solve for the archetypes and the representation coefficients. To integrate archetypal analysis with local regression or manifold learning, we propose to represent each data point as a convex combination of archetypes with further regularization enforcing that more weight is assigned to nearby archetypes. One way to achieve this is by selecting a fixed number of nearby archetypes. While simple, estimating the optimal number of archetypes is challenging (as it inherently depends on the nonlinear structure of the data) and the resulting model is not flexible. Another way to impose locality is by enforcing that the weights are sparse for which the well-known ℓ_0 minimization is a natural regularizer. Given that ℓ_0 minimization is intractable, a widely adopted technique is based on its convex relaxation which yields the ℓ_1 regularizer. However, since the weights are supported on the simplex, all the feasible solutions attain the same ℓ_1 norm.

In this paper, we propose K-*Deep Simplex (KDS)*, a unified optimization framework for local archetypal learning. In KDS, each data point $\mathbf{y} \in \mathcal{R}^d$ is expressed as a sparse convex combination of m atoms. These atoms define a dictionary $\mathbf{A} \in \mathcal{R}^{d\times m}$ to be learned from the data. To glean intrinsically low-dimensional manifold structure, we regularize to encourage representing a data point using nearby atoms. The proposed method learns a dictionary $\mathbf{A}$ and low-dimensional features with a structure imposed by convexity and locality of representation. To learn the atoms, we employ the alternating minimization framework which alternates between updating the atoms and updating the coefficients. The algorithm can also easily be mapped to a neural network architecture leading to interpretable neural networks. This mapping is along the lines of algorithm unrolling [18]–[21], an increasingly popular technique for structured deep learning.

A. Contributions

This paper introduces a structured dictionary learning model based on the idea of representing data as a convex combination of local archetypes. One immediate advantage of the method is that it leads to an interpretable framework. Since the coefficients are non-negative and sum to 1, they automatically enjoy a probabilistic interpretation.

Another advantage of the proposed algorithm is its connection to structured compressed sensing. We show that the proposed locality regularizer can be interpreted as a weighted ℓ_1 relaxation for a suitably defined ℓ_0 minimization. Under a certain generative model of data, we show how the proposed weighted ℓ_1 norm exactly recovers the underlying true sparse solution. In addition, for this generative model, we show stability of the weighted ℓ_1 norm. In contrast to the standard compressed sensing setting which depends on coherence and the restricted isometry property (which do not hold in our setting), our analysis hinges on intrinsic geometric properties of data.

The proposed locality regularizer is essentially a quadratic form of a Laplacian over a suitably defined graph. Since we learn a dictionary consisting of $m \ll n$ atoms, where m is independent of n and depends only on intrinsic geometric properties of the data, we show that the spectral embedding can be computed efficiently by only considering the $m \times m$ covariance matrix of the coefficient matrix.

We discuss the alternating minimization framework to solve the main optimization problem. We argue that in the typical setting where $m \ll n$, the proposed algorithm is scalable. In addition, since our KDS embedding can be computed efficiently, this naturally leads to a scalable spectral clustering algorithm.

We also map our iterative algorithm to a structured neural network. This mapping is along the lines of iterative algorithm unrolling [18]–[25] to solve our optimization problem. To be specific, we train a recurrent autoencoder with a nonlinearity that captures the constraint that our representation coefficients must lie on the probability simplex. To our knowledge, our use of algorithm unrolling for manifold learning is new.

For reproducibility, we will provide the code for all the experiments in this paper. To give a glimpse of the performance of KDS, Figure 1 shows the atoms the autoencoder learns for the classic two moons dataset and digits from the MNIST-5 dataset (5 digits from the MNIST dataset).

Differences from our prior work: Previous work in [26] by a subset of the authors of the present paper defines a weighted ℓ_0 norm and shows that the weighted ℓ_1 regularization studied in this paper recovers a unique solution under a certain generative model of data. Therein, we propose a simple alternating minimization algorithm to learn the sparse coefficients and the dictionary atoms and test it on two datasets. Some key differences between the work in [26] and the current work are summarized below:

1) Given fixed coefficients, we further consider the subproblem of minimizing over the dictionary. Our result is summarized in Theorem 4.
2) The weighted ℓ_0 norm defined in [26] is a useful definition if the sparsity is fixed. If the sparsity is not fixed, Theorem 1 in [26] is not correct and is not applicable. To fix this issue and have a theoretical result that does not depend on fixing the sparsity level, we define a more general weighted ℓ_0 norm in this paper (see Definition 5).
3) We compare our method to more baselines and consider more datasets (e.g., images of faces, hyperspectral data).
4) The main algorithm used in this paper is based on mapping the iterative algorithm to a neural network and departs from the previous algorithm which is based on alternating minimization.

We also note that parts of the current work have appeared in our previously unpublished paper [27]. In contrast to these prior works, the current work presents new theory, comparisons to more baselines, a detailed review of related work, and interpretations of the proposed regularizer.

B. Notation

Lowercase and uppercase boldface letters denote column vectors and matrices, respectively. We denote the Euclidean, ℓ_0, and ℓ_1 norms of a vector $\mathbf{x}$, respectively as $||\mathbf{x}||_2$, $||\mathbf{x}||_0$ and $||\mathbf{x}||_1$. The Frobenius and operator norm of a matrix $\mathbf{A}$ are respectively denoted as $||\mathbf{A}||_F$ and $||\mathbf{A}||$. $\langle \mathbf{x}, \mathbf{y}\rangle$ denotes the Euclidean inner product. $\langle \mathbf{A}\, , \mathbf{B}\rangle$ denotes the trace inner product. The vector $\mathbf{1}$ denotes a vector whose entries are all 1. $\Delta^p \equiv \{\mathbf{z} \in \mathcal{R}^p : \sum_{i=1}^p z_i = 1, \mathbf{z} \geqslant \mathbf{0}\}$ denotes the probability

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

simplex. Given a matrix $\mathbf{A}$, $\mathbf{a}_i$ denotes its i-th column. The set of $m \times n$ matrices where each column lies in the probability simplex Δ^m is denoted by S. $\text{diag}(\mathbf{x})$ represents a diagonal matrix whose entries are the vector $\mathbf{x}$. $\text{Tr}(\mathbf{A})$ denotes the trace of the matrix $\mathbf{A}$. The set of positive real numbers is denoted by $\mathcal{R}_+$. Given a scalar $\mathbf{x}_i$, $\mathbf{1}_{\mathcal{R}_+}(\mathbf{x}_i)$ denotes the indicator function whose value is 1 if $x_i > 0$ and is 0 otherwise. $\mathbf{e}_j$ denotes a vector of zeros except a 1 in the j-th position. $\sigma_{\max}(\mathbf{A})$ and $\sigma_{\min}(\mathbf{A})$ denote the largest and smallest singular values of $\mathbf{A}$.

II. Proposed Method: K-Deep Simplex

Let $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathcal{R}^{d\times n}$ be a set of n data points in $\mathcal{R}^d$. Our approach is to approximate each data point $\mathbf{y}_i$ by a convex combination of $m \ll n$ archetypes. We define a dictionary $\mathbf{A}$ which is a collection of the m archetypes, $\mathbf{A} = [\mathbf{a}_1, \ldots, \mathbf{a}_m] \in \mathcal{R}^{d\times m}$. For sake of presentation, we first consider the case where the data points can be represented exactly as a convex combination of the archetypes. This leads to $\mathbf{Y} = \mathbf{AX}$ where $\mathbf{X} = [\mathbf{x}_1, \ldots, \mathbf{x}_n] \in \mathcal{R}^{m\times n}$ is the coefficient or weight matrix. The convex combination implies $(\mathbf{x}_i)_j \geqslant 0$ for all i and j and $\mathbf{X}^\top \mathbf{1} = \mathbf{1}$. We note that this automatically provides us with a probabilistic interpretation of the coefficients. Next, we consider a suitable regularization with the aim that each data point is represented as a convex combination of its nearby archetypes. The regularization we consider is $\sum_{i,j} (\mathbf{x}_i)_j \|\mathbf{y}_i - \mathbf{a}_j\|^2$ where $(\mathbf{x}_i)_j$ denotes the j-th entry of $\mathbf{x}_i$. The resulting optimization program is given by

$$\begin{aligned} \min_{\substack{\mathbf{A}\in\mathcal{R}^{d\times m}\\ \mathbf{X}\in\mathcal{R}^{m\times n}}} \quad & \sum_{i,j} (\mathbf{x}_i)_j \|\mathbf{y}_i-\mathbf{a}_j\|^2 \\ \text{subject to} \quad & \mathbf{Y}=\mathbf{AX} \\ & \sum_j (\mathbf{x}_i)_j = 1 \quad \text{for } i=1,2,...,n \\ & (\mathbf{x}_i)_j \geqslant 0, \quad \text{for all } i,j. \end{aligned} \tag{1}$$

A. KDS interpretations

Below, we further explore the objective in the optimization program in (1) by discussing various interpretations. Note that, we focus on the locality regularization and do not consider the constraint $\mathbf{Y} = \mathbf{AX}$.

Graph matching: For a fixed $\mathbf{A}$, the objective in (1) can be related to graph matching. Consider a bipartite graph where the nodes are the data points and atoms. We consider matching the data points with the atoms using the coefficients to derive a cost matrix. Formally, we have the following

$$\min_{\mathbf{X}\in S} \sum_{i,j} (\mathbf{x}_i)_j \|\mathbf{y}_i - \mathbf{a}_j\|^2 = \min_{\mathbf{X}\in S} \ \text{Tr}(\mathbf{X}^T\mathbf{C}) = \min_{\mathbf{X}\in S} \langle \mathbf{X}, \mathbf{C}\rangle,$$

where $\mathbf{C} \in \mathcal{R}_+^{m\times n}$ denotes a cost matrix defined as $C_{ij} = ||\mathbf{y}_i - \mathbf{a}_j||_2^2$. The resulting problem is similar to the one to many graph matching problem [28].

Optimal transport: Given the set of points $\mathbf{Y}$ and $\mathbf{A}$, we define empirical measures $\mu_y = \frac{1}{n}\sum_{i=1}\delta_{\mathbf{y}_i}$ and $\mu_a = \frac{1}{m}\sum_{i=1}^m \delta_{\mathbf{a}_i}$ with δ denoting a Dirac measure. The squared Wasserstein-2 distance between the probability measures μ_y and μ_a is defined as

$$\mathcal{W}_2(\mu_y, \mu_a) = \min_{\gamma\in\Pi(\mu_y,\mu_a)} \sqrt{\sum_{i=1}^n \sum_{j=1}^m ||\mathbf{y}_i - \mathbf{a}_j||_2^2 \gamma_{ij}},$$

where γ is a joint probability measure over $\{\mathbf{y}_1, \ldots \mathbf{y}_n\} \times \{\mathbf{a}_1, ..., \mathbf{a}_m\}$ and $\Pi(\mu_y, \mu_a) = \{\gamma \in \mathcal{R}^{n\times m} | \gamma\mathbf{1} = \frac{1}{n}\mathbf{1}, \gamma^T\mathbf{1} = \frac{1}{m}\mathbf{1}\}$. If we let $\mathbf{X} = n\gamma^T$, the squared Wasserstein distance is equivalent to minimizing $\langle \mathbf{X}, \mathbf{C}\rangle$ over the set $\{\mathbf{X} \in \mathcal{R}_+^{n\times m} |\ \mathbf{X}^T\mathbf{1} = \mathbf{1}, \mathbf{X}\mathbf{1} = \frac{n}{m}\mathbf{1}\}$. In contrast to the standard regularizer which has a one-sided constraint (sum to 1 constraint as a result of convex combination), this new formulation further restricts the sum of coefficients across rows placing a hard limit on how often a given atom is used to represent data points.

K-means: Given data $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathcal{R}^{d\times n}$, the K-means problem seeks to simultaneously find m clusters with centers $\mathbf{A} = \{\mathbf{a}_1, ..., \mathbf{a}_m\}$ and assign each data point to one of the m clusters. The optimization problem is

$$\min_{\mathbf{C}\in\mathcal{R}^{n\times m}, \{\mathbf{a}_i\}_{i=1}^m} \sum_{j=1}^m \sum_{i=1}^n C_{ij} ||\mathbf{y}_i - \mathbf{a}_j||_2^2,$$

where $\mathbf{C} \in \{0,1\}^{n\times m}$ is a binary matrix satisfying $\forall i, \sum_{j=1}^m C_{ij} = 1$. The above minimization problem resembles the objective in (1). In fact, given $(\mathbf{X}^*, \mathbf{A}^*) = \operatorname*{arg\,min}_{\mathbf{A}\in\mathcal{R}^{d\times m}\ \mathbf{X}\in S} \sum_{i,j} (\mathbf{x}_i)_j \|\mathbf{y}_i - \mathbf{a}_j\|^2$, if each data point has a unique nearest atom in $\mathbf{A}^*$, it can be shown that each each column of $\mathbf{X}^*$ is one-sparse i.e., $\mathbf{X}^*$ is a binary assignment matrix.

Laplacian smoothness: We first define a set of vertices by combining the data points and atoms. The coordinate representation of the combined vertices is denoted by $\mathbf{R} = [\mathbf{Y}\ \ \mathbf{A}] \in \mathcal{R}^{d\times(n+m)}$. From this, we define a bipartite graph where edges only exist between data points and atoms i.e., in which an edge of weight $(\mathbf{x}_i)_j$ connects the vertex $\mathbf{y}_i$ and the vertex $\mathbf{a}_j$. The weight matrix $\mathbf{W} \in \mathcal{R}^{(n+m)\times(n+m)}$ is

$$\mathbf{W} = \left(\begin{array}{c|c} \mathbf{0} & \mathbf{X}^T \\ \hline \mathbf{X} & \mathbf{0} \end{array}\right). \tag{2}$$

The graph Laplacian is now defined as $\mathbf{L} = \mathbf{D} - \mathbf{W}$ where the diagonal degree matrix $\mathbf{D} \in \mathcal{R}^{(n+m)\times(n+m)}$ is defined as $D_{ii} = \sum_{j=1}^{n+m} W_{ij}$. We now show how the locality regularizer is connected to the quadratic form of the Laplacian.

Proposition 1. *Let $\mathbf{R} = [\mathbf{Y}\ \ \mathbf{A}] \in \mathcal{R}^{d\times(n+m)}$. Then,*

$$\sum_{i=1}^n \sum_{j=1}^m (\mathbf{x}_i)_j ||\mathbf{y}_i - \mathbf{a}_j||_2^2 = \text{Tr}(\mathbf{R}\mathbf{L}\mathbf{R}^T).$$

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

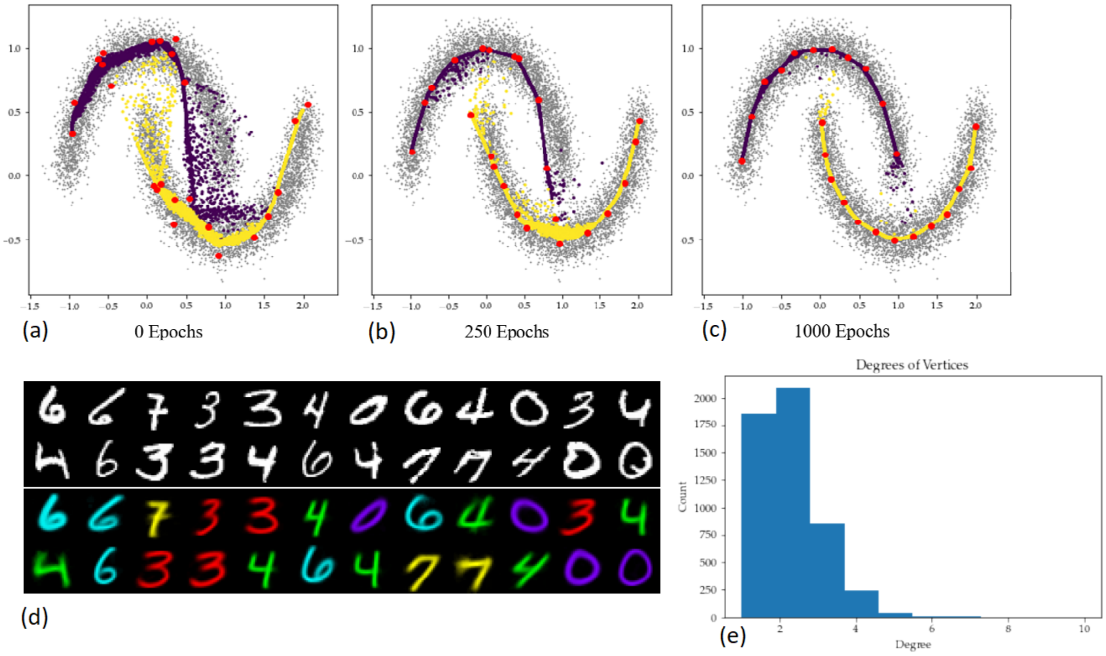

Fig. 1. (a-c) Training from a random initialization of atoms on the two moons data set. (d) A subset of the randomly initialized atoms for MNIST-5 (digits 0, 3, 4, 6, 7) before training (black and white) and after training and clustering (color). The number of data points is $n \approx 35000$ and the number of atoms is $m = 500$. (e) Degrees of vertices in the learned similarity graph. Despite being very sparse (most digits are represented using at most 5 atoms), the learned similarity graph retains enough information about the original data set that spectral clustering recovers these digits with 99% accuracy.

Proof.

$$
\begin{aligned}
&\sum_{i=1}^{n}\sum_{j=1}^{m}(\mathbf{x}_i)_j||\mathbf{y}_i-\mathbf{a}_j||_2^2\\
&=\sum_{i=1}^{n}\mathbf{y}_i^T\mathbf{y}_i\sum_{j=1}^{m}(\mathbf{x}_i)_j+\sum_{j=1}^{n}\mathbf{a}_j^T\mathbf{a}_j\sum_{i=1}^{n}(\mathbf{x}_i)_j-2\sum_{i,j}(\mathbf{x}_i)_j\mathbf{y}_i^T\mathbf{a}_j\\
&=\sum_{i=1}^{n}\mathbf{y}_i^T\mathbf{y}_i+\sum_{j=1}^{n}\mathbf{a}_j^T\mathbf{a}_j\,(\mathbf{X1})_j-2\sum_{i,j}(\mathbf{x}_i)_j\mathbf{y}_i^T\mathbf{a}_j\\
&=\mathrm{Tr}(\mathbf{Y}^T\mathbf{Y}\mathbf{I})+\mathrm{Tr}(\mathbf{A}^T\mathbf{A}\mathrm{diag}(\mathbf{X1}))-\mathrm{Tr}(\mathbf{R}^T\mathbf{R}\mathbf{W})\\
&=\mathrm{Tr}\left(\begin{bmatrix}\mathbf{Y}^T\mathbf{Y} & \mathbf{Y}^T\mathbf{A}\\ \mathbf{A}^T\mathbf{Y} & \mathbf{A}^T\mathbf{A}\end{bmatrix}\begin{bmatrix}\mathbf{I} & \mathbf{0}\\ \mathbf{0} & \mathrm{diag}(\mathbf{X1})\end{bmatrix}\right)-\mathrm{Tr}(\mathbf{R}^T\mathbf{R}\mathbf{W})\\
&=\mathrm{Tr}(\mathbf{R}^T\mathbf{R}\mathbf{D})-\mathrm{Tr}(\mathbf{R}^T\mathbf{R}\mathbf{W})\\
&=\mathrm{Tr}(\mathbf{R}^T\mathbf{R}(\mathbf{D}-\mathbf{W}))=\mathrm{Tr}(\mathbf{R}\mathbf{L}\mathbf{R}^T)
\end{aligned}
$$

□

Hence, the summation $\sum_{i=1}^{n}\sum_{j=1}^{m}(\mathbf{x}_i)_j||\mathbf{y}_i-\mathbf{a}_j||_2^2$ is precisely the *Laplacian quadratic form* of the graph whose vertices are the data points and atoms where the weight function is the representation coefficients.

III. Related Works

One of the goals of the proposed model is to combine manifold learning with sparse coding/dictionary learning. To our knowledge, the first work that integrates sparse coding, manifold learning, and slow feature analysis is the sparse manifold transform framework proposed in [29]. Therein, non-linear sparse coding using a learned dictionary is first used to map the data into a high-dimensional space. The next step extracts low-dimensional representations employing a matrix learned using a framework known as functional embedding [29]. In this paper, the aim is a combination of linear sparse coding and dictionary learning. In addition, our analysis focuses on structured dictionaries coming from triangulation of a set of points. Below, we review related works in dictionary learning, manifold learning, and non-negative matrix factorization.

A. Locality constrained dictionary learning

Our work connects with sparse coding [30] and dictionary learning. In sparse coding, given a fixed *dictionary* $\mathbf{A} \in \mathcal{R}^{d\times m}$ of m atoms, a data point $\mathbf{y} \in \mathcal{R}^d$ is represented as a linear combination of at most $k \ll m$ columns of $\mathbf{A}$. The dictionary $\mathbf{A}$ can be predefined [31] (e.g., Fourier bases, wavelets, curvelets) or adaptively learned from the data [32]–[35]. The latter setting where the dictionary is simultaneously estimated with the sparse coefficients is the standard dictionary learning problem. We consider the prototypical form of the optimization objective for dictionary learning $\sum_{i=1}^{n}\frac{1}{2}||\mathbf{y}_i-\mathbf{A}\mathbf{x}_i||_2^2+R(\mathbf{x}_i,\mathbf{A},\mathbf{y}_i)$ where $R(\mathbf{x}_i,\mathbf{y}_i,\mathbf{A})$ is a regularization term on the representation coefficients, the dictionary atoms and the data points.

In Table I, we review related works in graph regularized coding and locality constrained coding. The main idea in these works is to employ a Laplacian smoothness regularization such that if two data points are close, the regularization encourages their coefficients to be similar [36], [37]. A few remarks are in order in how KDS compares to these methods. First, in KDS regularization, the underlying graph is not fixed but iteratively updated since the weights of the graph depend on the sparse representation coefficients. This is in contrast to methods that consider trace$(\mathbf{X}\mathcal{L}\mathbf{X}^T)$ where $\mathcal{L}$ is a priori fixed based on

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

TABLE I
RELATED WORK

Work	$R(\mathbf{X}, \mathbf{Y}, \mathbf{A})$	Notes on constraints
[45]	trace$(\mathbf{X}\mathcal{L}\mathbf{X}^T) + \lambda\|\|\mathbf{X}\|\|_1$	Sparse $\mathbf{X}$ and $\|\|\mathbf{a}_i\|\|_2^2 \leqslant c$ $\mathcal{L}$ priori fixed
[46]	None	Simplex constraints on X
[38]	trace$(\mathbf{X}\mathcal{L}\mathbf{X}^T)$	$\mathcal{L}$ priori fixed
[40]	$\sum_{i,j}(\mathbf{x}_i)_j^2 \exp\left(\frac{\|\|\mathbf{y}_i-\mathbf{a}_j\|\|}{\sigma}\right)$	$\mathbf{X}^T\mathbf{1} = \mathbf{1}$ and $\|\|\mathbf{a}_i\|\|_2^2 \leqslant c$
[47]	$\sum_{i,j}(\mathbf{x}_i)_j^2\|\|\mathbf{y}_i - \mathbf{a}_j\|\|^2 + \lambda\|\|\mathbf{X}\|\|_F^2$	$\mathbf{X}^T\mathbf{1} = \mathbf{1}$, $(\mathbf{x}_i)_j$ set to zero based on neighborhood
[48]	trace$(\mathbf{X}\mathcal{L}\mathbf{X}^T) + \lambda\|\|\mathbf{X}\|\|_0$	Sparse $\mathbf{X}$, $\mathcal{L}$ priori fixed
[39]	$\sum_{i,j} \|(\mathbf{x}_i)_j\| \, \|\|\mathbf{y}_i - \mathbf{a}_j\|\|^{1+p}$	$\mathbf{X}^T\mathbf{1} = \mathbf{1}$.
[49]	trace$(\mathbf{X}^T\mathcal{L}\mathbf{X})$	$\|\|\mathbf{a}_i\|\|^2 = 1$. An additional SVM regularization
[43]	$\sum_{i,j}(\mathbf{x}_i)_j \, \|\|\mathbf{y}_i - \mathbf{y}_j\|\|^2 + \|\|\mathbf{X}\|\|_F^2$ diag$(\mathbf{X}) = \mathbf{0}$	Simplex constraints, No dictionary learning
[41]	$\sum_{i,j} Q_{ij}(\mathbf{x}_i)_j$	$\mathbf{X}^T\mathbf{1} = \mathbf{1}$ $\mathbf{Q} \equiv$ proximity regularizer

similarity of the data points. In Table I, the closest methods to KDS are [38]–[41]. However, the coefficients in these methods do not lie on the simplex and the regularizers are based on $(\mathbf{x}_i)_j^2$ or $|(\mathbf{x}_i)_j|$. The implication of these choices is that the sparse coding step in [38], [40] yields a unique solution. This departs from our setup where the sparse coding step in general does not have a unique solution. In addition, the aforementioned works lack theoretical analysis that shows that the sparse coding step provably results a sparse solution. The sparse manifold clustering and embedding algorithm (SMCE) [41] employs proximity regularization that promotes representation using local dictionaries. A drawback of SMCE is its computational inefficiency since the dictionary is essentially all the data points. Focusing on the problem of clustering, the work in [42] introduces an optimization framework aimed at jointly learning a union-of-subspace representation and performing clustering. In this manuscript, the optimization objective retains a broad scope, learning representations that are not tailored to a specific end task. Finally, the work in [43] proposes a similar regularization to ours with the authors referring to it as "adaptive distance regularization". However, the methodology therein is based on using the data matrix as a dictionary and lacks theoretical analysis. Finally, we refer the reader to [44] to find a comprehensive overview of nonlinear manifold clustering algorithms.

B. Manifold learning

Our setup is along the lines of methods that learn local or global features of data using neighborhood analysis. For instance, locally linear embedding (LLE) [13] provides a low dimensional embedding using weights that are defined as the reconstruction coefficients of data points from their neighbors. The choice of the optimal neighborhood size is important for LLE as it determines the features obtained and subsequently the performance of downstream tasks. Geometric multiresolution analysis (GMRA) is a fast and efficient algorithm that learns multiscale representations of the data based on local tangent space estimations [34], [35]. Since the dictionary elements used to reconstruct are defined locally, GMRA is not immediately useful for global downstream tasks, e.g., clustering. We also note that the work in [50] develops a theoretical framework for regression on low-dimensional sets embedded in high dimensions. The regression is done via local polynomial fitting which resembles local convex approximation in KDS albeit the former method is applied to the supervised setting.

C. Scalable manifold learning via landmarks

For large datasets, the embedding step in manifold learning techniques which typically involves a spectral problem can be costly. One approach to circumvent the computational challenge is based on finding an approximate solution by first identifying a subset of points designated as landmarks or exemplars. For instance, the works in [51], [52] propose landmark isometric feature mapping (Isomap) and landmark multidimensional scaling (MDS) which are respectively scalable versions of Isomap [12] and classical MDS [53]–[55]. The work in [56] first considers sparse coding (assuming pre-computed m landmarks) of all data points to obtain a sparse representation matrix $\mathbf{Z} \in \mathcal{R}^{m \times n}$. It then obtains spectral embeddings using the right singular vectors of a scaled $\mathbf{Z}$. Another approach along the lines of our work is the work in [57] which proposes an efficient version of the locally linear embedding method using landmarks. In contrast to our approach which learns the landmarks, we note that the methods in [56], [57] identify the landmarks from the full data using strategies such as random sampling and clustering. A method inspired by LLE for semi-supervised learning, local anchor embedding (LAE), is proposed in [58]. In this approach, the anchors are centers learned from the K-means algorithm. To obtain the representation coefficient of each data point, LAE solves a least squares problem in a dictionary of s-nearest anchors and with coefficients restricted on the simplex. Compared to our approach, the anchor learning step is disjoint from the sparse coding step in LAE. In addition, while LAE introduces sparsity by setting number of nearest anchors, our approach is based on promoting sparsity via a flexible proximity regularization. There are scalable landmark/exemplar methods for sparse subspace clustering e.g., [59]–[61] but subspace clustering stipulates global affine structure that is not directly applicable to the general case of nonlinear manifolds.

D. Non-negative matrix factorization

Non-negative matrix factorization (NMF) considers the problem of approximating a nonnegative data matrix using underlying components that are also non-negative [62], [63]. Let $\mathcal{R}_{\geqslant 0}^{m \times n}$ denote the set of $m \times n$ nonnegative matrices. Given a data matrix $\mathbf{Y} \in \mathcal{R}_{\geqslant 0}^{d \times n}$, approximate NMF seeks non-negative matrices $\mathbf{W} \in \mathcal{R}_{\geqslant 0}^{d \times m}$ and $\mathbf{H} \in \mathcal{R}_{\geqslant 0}^{m \times n}$ that best approximate the data. Choosing the Euclidean distance as a loss function, the problem can be formulated as $\min_{\mathbf{W} \in \mathcal{R}_{\geqslant 0}^{d \times m}, \mathbf{H} \in \mathcal{R}_{\geqslant 0}^{m \times n}} ||\mathbf{Y} - \mathbf{W}\mathbf{H}||_F^2$. Different models on NMF put forth various conditions on the data matrix and the components. The work in [64] proposes a convex-model for NMF for a general data matrix with the restriction that $\mathbf{H}$ is non-negative and the columns of $\mathbf{W}$ lie in the column space of $\mathbf{Y}$. A similar work to ours is in [65] where the authors propose simplex structured matrix factorization (SSMF) which considers the recovery of $\mathbf{W}$ and $\mathbf{H}$ given a

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

generic data matrix with the restriction that $\mathbf{H} \in S$. Therein, the authors show that the exact $\mathbf{W}$ can be recovered by considering a maximum volume ellipsoid inscribed in the convex hull of the data points. We note that the model assumption in [65] assumes a full column rank $\mathbf{A}$ and a full row rank $\mathbf{X}$ which we do not assume in our setting. Further discussion of different assumptions for identifiability of SSMF can be found in [66]. Finally, the works in [67] and [68] in hyperspectral imagery study a similar problem as ours but with the difference that the former considers a non-negative constraint and the latter uses the ℓ_0 regularizer on the simplex.

IV. Theoretical analysis

To solve the optimization program in (1), a common approach is alternating minimization which is comprised of two steps. The first step is sparse coding and the second step is dictionary learning. In this section, we provide theoretical analysis for the sparse coding and dictionary learning steps of our proposed optimization program in (1). The sparse coding problem fixes $\mathbf{A}$ and optimizes over $\mathbf{X}$ while the dictionary learning problem fixes $\mathbf{X}$ and optimizes for $\mathbf{A}$. We also discuss how to obtain a low-dimensional embedding of data points. Part of this analysis was completed in our prior work in [26].

A. Sparse coding

The theoretical analysis for the sparse coding step assumes a specific model for the atoms and for generating the data points. Before describing the model, we start with essential background information on d-simplices, triangulations and a Delaunay triangulation.

Definition 1. *A d-simplex is the convex hull of a set of $d+1$ points $\{\mathbf{a}_0, \mathbf{a}_1, .., \mathbf{a}_d\}$ in $\mathcal{R}^d$.*

For example, a 0-simplex and 1-simplex respectively correspond to a point and a line segment. The $d+1$ points that determine the d-simplex are called vertices of the simplex. Next, we define the s-face of a d-simplex. The definition is restated from [69].

Definition 2. *An s-face of a simplex is the convex combination of a subset of $s+1$ vertices of the simplex.*

For example, a 0-face corresponds to a point, a 1-face is an edge and a 2-face is a triangular facet. The next definition concerns triangulation given a set of points. For the purposes of our analysis, we use the following definition [69], [70].

Definition 3. *Given a set of points $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$ in $\mathcal{R}^d$, a triangulation T is a set of d-simplices that partition the convex hull of $\mathbf{P}$ such that the intersection of any two simplices in T is either empty or a common face.*

We now proceed to define the main object of our theoretical analysis, the Delaunay triangulation.

Definition 4. *A Delaunay triangulation of a set of m points $\mathbf{P} = \{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$ in $\mathcal{R}^d$, DT$(\mathbf{P})$, is any triangulation of $\mathbf{P}$ such that for every d-simplex in DT$(\mathbf{P})$, the circumscribing hypersphere of the d-simplex does not contain any other point of $\mathbf{P}$.*

Given a set of points in $\mathcal{R}^d$, the existence of a unique Delaunay triangulation is based on the following geometric condition: the affine span of $\mathbf{P}$ is d-dimensional and no $d+2$ points of $\mathbf{P}$ lie on the same sphere. We refer to such points as points in a *general position*.

a) Model for generating atoms and data: We consider m landmark points $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m$ in $\mathcal{R}^d$ with $m \geqslant d+1$ in general position meaning that there is a unique Delaunay triangulation. Each data point is in the convex hull of the m landmark points. Figure 2 illustrates the model when $d=2$.

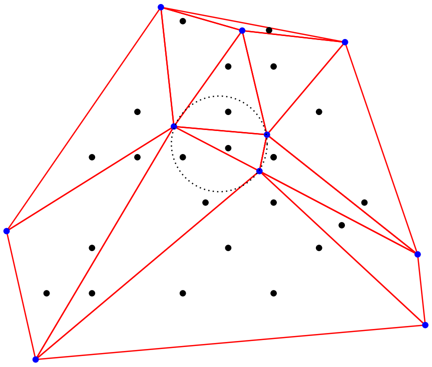

Fig. 2. The blue dots indicate the atoms which generate the data points. Each black dot, denoting a data point, is a convex combination of three atoms which are vertices of the triangle the point belongs to. Note that the circumscribing circle of any triangle does not contain any additional landmark points.

b) Sparse coding under the Delaunay triangulation model: Let $\mathbf{A} \in \mathcal{R}^{d \times m}$ be the dictionary of the m landmarks defined as $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m]$. Any point $\mathbf{y}$ in the convex hull of the m landmarks can be written as $\mathbf{y} = \mathbf{A}\mathbf{x}$ where $\mathbf{x} \in \Delta^m$. However, note that there may be multiple ways to represent the point $\mathbf{y}$ as a convex combination of the landmark points. Since our aim is to obtain sparse representations, we focus on the problem of finding a unique sparse solution to $\mathbf{y} = \mathbf{A}\mathbf{x}$. Let DT$(\mathbf{A})$ denote the set of d-simplices that constitute the Delaunay triangulation of $\{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m\}$. For our setting, we define the sparsest representation to be the representation of a point $\mathbf{y}$ using the vertices of the d-simplex of DT$(\mathbf{A})$ it belongs to. As an example, if $d=2$, this will be representing the point using the vertices of the triangle it belongs to. This motivates the following definition of a weighted ℓ_0 pseudo-norm.

Definition 5. *Assume m landmark points $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m$ in $\mathcal{R}^d$ have a unique Delaunay triangulation DT$(\mathbf{A})$. Let $\mathbf{y} \in \mathcal{R}^d$ be an interior point of a d-simplex of DT$(\mathbf{A})$ with circumcenter $\mathbf{c}$. The* weighted ℓ_0 norm *of $\mathbf{x}$ is defined as*

$$\ell_{w,0}(\mathbf{x}) = \frac{1}{\|\mathbf{x}\|_0} \sum_{i=1}^{m} \mathbf{1}_{\mathcal{R}_+}(x_i) ||\mathbf{c} - \mathbf{a}_i||^2, \qquad (3)$$

where $\mathbf{1}_{\mathcal{R}_+}(x_i) = 1$ if $x_i > 0$ and 0 otherwise.

Given the above definition of a weighted ℓ_0 norm and the fact that a given point $\mathbf{y}$ admits different representations as a convex combination of the dictionary atoms, the natural question is the sense in which this norm is minimal i.e., among the different representations, which ones admit minimal values in this norm? The next theorem shows that the local reconstruction is minimal

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

in the weighted ℓ_0 norm. The result of Theorem 1 follows from the following lemma, which we prove below.

Lemma 1. *Let $\mathbf{A} \in \mathcal{R}^{d\times m}$ be the dictionary of the m landmarks defined as $\mathbf{A} = [\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m]$. Let DT$(\mathbf{A})$ denote a set of d-simplices of the Delaunay triangulation of $\{\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m\}$. If f is a d-simplex of DT$(\mathbf{A})$ defined by the vertices $\{\mathbf{a}_j : j \in T, |T| = d+1\}$, there is a hypersphere with center $\mathbf{c}$ and radius R such that $||\mathbf{a}_j - \mathbf{c}|| = R$ if $j \in T$ and $||\mathbf{a}_j - \mathbf{c}|| > R$ if $j \notin T$.*

Proof. f is a d-simplex of DT$(\mathbf{A})$ where the indices of its vertices are in T. Let $\mathbf{c}$ and R respectively denote the center and radius of the circumscribing hypersphere of f. By construction, $||\mathbf{a}_j - \mathbf{c}|| = R$ if $j \in T$. For contradiction, assume that there is a $j \notin T$ such that $||\mathbf{a}_j - \mathbf{c}|| \leqslant R$. This contradicts the definition of a Delaunay triangulation in Definition 4 since $\mathbf{a}_j$ will be an interior point of the circumscribing hypersphere.

□

Theorem 1. *Given a set of landmarks $\{\mathbf{a}_1, \dots, \mathbf{a}_m\}$ with a unique Delaunay triangulation DT$(\mathbf{A})$, let $\mathbf{y} \in \mathcal{R}^d$ be an interior point of the d-simplex of DT$(\mathbf{A})$ with circumcenter $\mathbf{c}$ and radius R. Let*

$$\mathbf{x}^* = \underset{\mathbf{x}\in\Delta^m}{\arg\min}\ \ell_{w,0}(\mathbf{x}) \quad \textit{s.t.} \quad \mathbf{y} = \mathbf{A}\mathbf{x}.$$

Then, $\mathbf{x}^*$ is such that $\{j : \mathbf{x}_j^* \neq 0\}$ correspond to the indices of the vertices of the d-simplex of DT$(\mathbf{A})$ that contains $\mathbf{y}$.

Proof. Consider a d-simplex of DT$(\mathbf{A})$ containing $\mathbf{y}$ defined by the vertices $\{\mathbf{a}_j : j \in T, |T| = d+1\}$. Using vertices in T, $\mathbf{y}$ can be represented as a convex combination using coefficient vector $\mathbf{x}^*$. Let $\mathbf{x}$ be another feasible solution of the program with support T'. We now apply Lemma 1 to obtain

$$\begin{aligned}\frac{1}{||\mathbf{x}||_0}\sum_{i\in T'} \mathbf{1}_{\mathcal{R}_+}(x_i)||\mathbf{c}-\mathbf{a}_i||^2 &> R^2 \sum_{i\in T'} \frac{\mathbf{1}_{\mathcal{R}_+}(x_i)}{||\mathbf{x}||_0} \\ &= R^2 \\ &= R^2 \sum_{i\in T} \frac{\mathbf{1}_{\mathcal{R}_+}(x_i^*)}{||\mathbf{x}^*||_0} \\ &= \frac{1}{||\mathbf{x}^*||_0}\sum_{i\in T} \mathbf{1}_{\mathcal{R}_+}(x_i^*)||\mathbf{c}-\mathbf{a}_i||^2\end{aligned}$$

Therefore, the sparse representation using the vertices in T is the optimal solution to the $\ell_{w,0}$ minimization problem. □

Given a reconstruction $\mathbf{y} = \mathbf{A}\mathbf{x}$, we note that the weighted ℓ_0 norm puts a uniform prior on all atoms which are used in the representation. However, there are two drawbacks of this regularization. First, the definition of the weighted ℓ_0 norm depends on knowing the circumcenter of the d-simplex the point belongs to. In addition, the regularizer uses an indicator function which is not suitable for optimization. To obtain a regularization amenable to optimization, we now define a convex relaxation of the weighted ℓ_0 problem as follows.

Definition 6. *Assume m landmark points $\mathbf{a}_1, \mathbf{a}_2, ..., \mathbf{a}_m$ in $\mathcal{R}^d$ with a unique Delaunay triangulation DT$(\mathbf{A})$. Let the point $\mathbf{y} \in \mathcal{R}^d$ be in the convex hull of the landmark points i.e., $\mathbf{y} = \mathbf{A}\mathbf{x}$ with $\mathbf{x} \in \Delta^m$. The* weighted ℓ_1 *norm of $\mathbf{x}$ is defined as*

$$\ell_{w,1}(\mathbf{x}) = \sum_{i=1}^{m} x_i\, ||\mathbf{y} - \mathbf{a}_i||^2, \tag{4}$$

Analogous to compressed sensing theory, the next question is the sense in which a weighted ℓ_1 minimization is equivalent to a weighted ℓ_0 minimization problem. This equivalency is summarized in Theorem 2. The following lemma will be essential to the proof of Theorem 2.

Lemma 2. *Given the dictionary of landmarks $\mathbf{A} \in \mathcal{R}^{d\times m}$, let $\mathbf{y} = \mathbf{A}\mathbf{x}$ for $\mathbf{x} \in \Delta^m$. For any arbitrary point $\mathbf{c} \in \mathcal{R}^d$,*

$$\ell_{w,1}(\mathbf{x}) = \sum_{i=1}^{m} x_i\, ||\mathbf{y} - \mathbf{a}_i||^2 = \sum_{i=1}^{m} x_i||\mathbf{a}_i - \mathbf{c}||^2 - ||\mathbf{y} - \mathbf{c}||^2.$$

Proof. We expand $\ell_{w,1}(\mathbf{x})$ as follows and use the fact that $\mathbf{y} = \mathbf{A}\mathbf{x}$ and $\mathbf{x} \in \Delta^m$:

$$\begin{aligned}&\sum_{i=1}^{m} x_i\, ||\mathbf{y} - \mathbf{a}_i||^2 \\ &= \sum_{i=1}^{m} x_i\, ||(\mathbf{y} - \mathbf{c}) + (\mathbf{c} - \mathbf{a}_i)||^2 \\ &= \sum_{i=1}^{m} x_i\, \left(||\mathbf{y} - \mathbf{c}||^2 + ||\mathbf{a}_i - \mathbf{c}||^2 - 2(\mathbf{y} - \mathbf{c})^T(\mathbf{a}_i - \mathbf{c})\right) \\ &= ||\mathbf{y} - \mathbf{c}||^2 + \sum_{i=1}^{m} x_i||\mathbf{a}_i - \mathbf{c}||^2 - 2(\mathbf{y} - \mathbf{c})^T \left(\sum_{i=1}^{m} x_i\mathbf{a}_i - \mathbf{c}\right) \\ &= ||\mathbf{y} - \mathbf{c}||^2 + \sum_{i=1}^{m} x_i||\mathbf{a}_i - \mathbf{c}||^2 - 2||\mathbf{y} - \mathbf{c}||_2^2 \\ &= \sum_{i=1}^{m} x_i||\mathbf{a}_i - \mathbf{c}||^2 - ||\mathbf{y} - \mathbf{c}||^2.\end{aligned}$$

□

Theorem 2. *Given a set of landmarks $\{\mathbf{a}_1, \dots, \mathbf{a}_m\}$ with a unique Delaunay triangulation DT$(\mathbf{A})$, let $\mathbf{y} \in \mathcal{R}^d$ be an interior point of a d-simplex of DT$(\mathbf{A})$. Let*

$$\mathbf{x}^* = \underset{\mathbf{x}\in\Delta^m}{\arg\min} \sum_i x_i||\mathbf{y} - \mathbf{a}_i||^2 \quad \textit{s.t.} \quad \mathbf{y} = \mathbf{A}\mathbf{x}.$$

Then, $\mathbf{x}^*$ is such that $\{i : \mathbf{x}_i^* \neq 0\}$ correspond to the indices of the vertices of the d-simplex of DT$(\mathbf{A})$ that contains $\mathbf{y}$.

Proof. Consider the d-simplex containing $\mathbf{y}$ defined by the vertices $\{\mathbf{a}_j : j \in T, |T| = d+1\}$. Since $\mathbf{y}$ is an interior point of the d-simplex, it can be represented as a convex combination of its vertices using coefficient vector $\mathbf{x}^*$. Note that $\mathbf{x}^*$ is supported on T with $||\mathbf{x}^*||_0 = d+1$. Let $\mathbf{x}$ be another feasible solution of the program with support T'. We now apply Lemma 3 to $\mathbf{y} = \mathbf{A}\mathbf{x}$ with $\mathbf{c}$ as the circumcenter of the d-simplex that contains $\mathbf{y}$:

$$\sum_{j\in T'} x_j||\mathbf{y} - \mathbf{a}_j||^2 = \sum_{j\in T'} x_j||\mathbf{a}_j - \mathbf{c}||^2 - ||\mathbf{y} - \mathbf{c}||^2.$$

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

We now apply Lemma 1 to lower bound the above term. Specifically, we use the fact that $||\mathbf{a}_j - \mathbf{c}|| > R$ if $j \notin T$ and $||\mathbf{a}_j - \mathbf{c}|| = R$ if $j \in T$:

$$\begin{aligned}\sum_{j\in T'} x_j\|\mathbf{y}-\mathbf{a}_j\|^2 &= \sum_{j\in T'} x_j||\mathbf{a}_j-\mathbf{c}||^2 - \|\mathbf{y}-\mathbf{c}\|^2.\\ &> \sum_{j\in T'} x_j R^2 - \|\mathbf{y}-\mathbf{c}\|^2.\\ &= R^2 - \|\mathbf{y}-\mathbf{c}\|^2.\\ &= \sum_{j\in T} x_j^*||\mathbf{a}_j-\mathbf{c}||^2 - \|\mathbf{y}-\mathbf{c}\|^2.\\ &= \sum_{j\in T} x_j^*\|\mathbf{y}-\mathbf{a}_j\|^2.\end{aligned}$$

Above, the inequality in the second line uses the fact that there is at least one index in T' that is not in T. The last equality follows from applying Lemma 3 with $\mathbf{y} = \mathbf{A}\mathbf{x}^*$ and $\mathbf{c}$ as the circumcenter. We have established that $\ell_{w,1}(\mathbf{x}) > \ell_{w,1}(\mathbf{x}^*)$ for any feasible $\mathbf{x}$. Therefore, the sparse representation using the vertices in T is the optimal solution to the $\ell_{w,1}$ minimization problem.

□

B. Stability analysis

In this section, we consider the stability of sparse representations when an input data is perturbed by a bounded additive noise. Formally, given a data point $\mathbf{y} \in \mathcal{R}^d$, the data is perturbed resulting $\tilde{\mathbf{y}}$ with the condition that $||\mathbf{y} - \tilde{\mathbf{y}}||_2 \leqslant \varepsilon$. In the analysis to follow, the notion of a local dictionary is used which we define below.

Definition 7. *Given a set of landmarks $\{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$ with a unique Delaunay triangulation DT($\mathbf{A}$), let $\mathbf{y} \in \mathcal{R}^d$ be interior points of the d-simplex of DT($\mathbf{A}$). Then the local dictionary $\mathbf{A}_L \in \mathcal{R}^{d\times d+1}$ associated to $\mathbf{y}$ is $\mathbf{A}_L = [\mathbf{a}_{j_1}\, \mathbf{a}_{j_2} \ldots \mathbf{a}_{j_{d+1}}]$, where the indices $\{j_k\}_{k=1}^{d+1}$ correspond to the vertices of the d-simplex that contains $\mathbf{y}$.*

The utility of a local dictionary is that it allows us to express a data point in terms of its barycentric coordinates.

Definition 8. *Given a local dictionary $\mathbf{A}_L \in \mathcal{R}^{d\times(d+1)}$ associated to $\mathbf{y} \in \mathcal{R}^d$, the barycentric coordinates of $\mathbf{y}$ is the unique solution to the linear system $\mathbf{B}_L\mathbf{x} = \mathbf{z}$, where $\mathbf{B}_L \in \mathcal{R}^{(d+1)\times(d+1)}$ is defined as $\mathbf{B}_L = \begin{pmatrix}\mathbf{A}_L\\ \mathbf{1}_d\end{pmatrix}$ and $\mathbf{z} = \begin{pmatrix}\mathbf{y}\\ 1\end{pmatrix}$.*

Theorem 3. *Given a set of landmarks $\{\mathbf{a}_1, \ldots, \mathbf{a}_m\}$ with a unique Delaunay triangulation DT($\mathbf{A}$), let $\mathbf{y}, \tilde{\mathbf{y}} \in \mathcal{R}^d$ be interior points of the same d-simplex of DT($\mathbf{A}$). Further, assume that $||\mathbf{y} - \tilde{\mathbf{y}}|| \leqslant \varepsilon$ and $\mathbf{y} = \mathbf{A}\mathbf{x}^*$ where*

$$\mathbf{x}^* = \underset{\mathbf{x}\in\Delta^m}{\arg\min} \sum_j x_j\|\mathbf{y}-\mathbf{a}_j\|^2 \quad \textit{s.t.} \quad \mathbf{y} = \mathbf{A}\mathbf{x}.$$

Let $\tilde{\mathbf{x}}^*$ be the optimal solution to the following $\ell_{w,1}$ minimization problem.

$$\tilde{\mathbf{x}}^* = \underset{\mathbf{x}\in\Delta^m}{\arg\min} \sum_j x_j\|\tilde{\mathbf{y}}-\mathbf{a}_j\|^2 \quad \textit{s.t.} \quad \tilde{\mathbf{y}} = \mathbf{A}\mathbf{x}.$$

Then, $\tilde{\mathbf{x}}^*$ to the above program is such that

$$||\tilde{\mathbf{x}}^* - \mathbf{x}^*|| \leqslant \frac{1}{\sigma_{\min}(\mathbf{B}_L)}\,\varepsilon,$$

where $\mathbf{B}_L = \begin{pmatrix}\mathbf{A}_L\\ \mathbf{1}_d\end{pmatrix}$ and $\mathbf{A}_L$ is the local dictionary associated to $\tilde{\mathbf{y}}$.

Proof. Since $\mathbf{y}$ and $\tilde{\mathbf{y}}$ belong to the same simplex of DT($\mathbf{A}$), they have the same local dictionary denoted by $\mathbf{A}_L$. Using Theorem 4, the optimal solution $\tilde{\mathbf{x}}^*$ is such that it is only nonzero on the indices corresponding to vertices of the simplex that contains $\tilde{\mathbf{y}}$. An analogous argument could be made for $\mathbf{x}^*$. It then follows that $\mathbf{y} = \mathbf{A}_L\mathbf{x}^*$ and $\tilde{\mathbf{y}} = \mathbf{A}_L\tilde{\mathbf{x}}^*$. In what follows, we form a square linear system by considering an additional constraint that the coefficients must sum to 1. To that end, we define $\mathbf{z}, \tilde{\mathbf{z}} \in \mathcal{R}^{d+1}$ as follows: $\mathbf{z} = \begin{pmatrix}\mathbf{y}\\ 1\end{pmatrix}$ and $\tilde{\mathbf{z}} = \begin{pmatrix}\tilde{\mathbf{y}}\\ 1\end{pmatrix}$. Note that $||\mathbf{z}-\tilde{\mathbf{z}}||_2 = ||\mathbf{y}-\tilde{\mathbf{y}}||_2$. Further, $\mathbf{z} = \mathbf{B}_L\mathbf{x}^*$ and $\tilde{\mathbf{z}} = \mathbf{B}_L\tilde{\mathbf{x}}^*$. We proceed to lower bound $||\mathbf{z} - \tilde{\mathbf{z}}||_2$:

$$||\mathbf{z} - \tilde{\mathbf{z}}||_2 = ||\mathbf{B}_L(\mathbf{x}^* - \tilde{\mathbf{x}}^*)||_2 \geqslant \sigma_{\min}(\mathbf{B}_L)||\mathbf{x}^* - \tilde{\mathbf{x}}||_2,$$

where $\sigma_{\min}(\mathbf{B}_L) > 0$ (this follows from the assumption that the landmarks are in general position). Combining this lower bound with $||\mathbf{y} - \tilde{\mathbf{y}}||_2 \leqslant \varepsilon$, we obtain

$$||\tilde{\mathbf{x}}^* - \mathbf{x}^*|| \leqslant \frac{1}{\sigma_{\min}(\mathbf{B}_L)}\,\varepsilon.$$

□

Remark: We would like to highlight that the affine constraint on the coefficients ensures that the aforementioned theorem remains valid even when the data points are translated. However, the stability of the theorem is contingent upon the minimum singular value of a shifted $\mathbf{B}_L$. We note that when the noise is sufficiently low and $\sigma_{\min}(\mathbf{B}_L)$ is appropriately large, the stability analysis ensures a robust sparse solution. This robustness depends upon the magnitude of $\sigma_{\min}(\mathbf{B}_L)$, which in turn is influenced by the geometrical structure of the localized dictionary. Initial numerical experiments suggest that if the localized dictionaries are “well-structured”, $\sigma_{\min}(\mathbf{B}_L)$ tends to be relatively large, whereas smaller values of $\sigma_{\min}(\mathbf{B}_L)$ correspond to elongated triangles. Details on the numerical experiments can be found in the Supplementary Materials.

C. Optimal dictionary

In the theoretical analysis so far, we have studied the problem of recovering a sparse coefficient vector given a fixed dictionary. In this section, we assume that the sparse coefficients are fixed and study the optimization problem over the dictionary. In particular, we study the optimal solution defined as follows.

$$\mathbf{A}^* = \underset{\mathbf{A}\in\mathcal{R}^{d\times m}}{\arg\min}\ ||\mathbf{Y}-\mathbf{A}\mathbf{X}||_F^2 + \lambda\sum_{i=1}^{n}\sum_{j=1}^{m}(\mathbf{x}_i)_j||\mathbf{y}_i-\mathbf{a}_j||^2, \quad (5)$$

where $\lambda > 0$ is a regularization parameter. Below, we will prove that $\mathbf{A}^*$ is unique and has a closed form solution.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

Theorem 4. *For fixed $\mathbf{X} \in S$, $\mathbf{A}^*$ is given by*

$$\mathbf{A}^* = (1+\lambda)\mathbf{Y}\mathbf{X}^T\mathbf{H}^{-1},$$

where $\mathbf{H} = \mathbf{X}\mathbf{X}^T + \lambda diag(\mathbf{X}\mathbf{1})$.

Proof. Let $f(\mathbf{A})$ denote the objective function in (5). The proof of the theorem relies on showing that $f(\mathbf{A})$ is strongly convex. Some calculation yields $\nabla f(\mathbf{A}) = 2(\mathbf{A}\mathbf{X} - \mathbf{Y})\mathbf{X}^T + 2\lambda\mathbf{A}\text{diag}(\mathbf{X}\mathbf{1}) - 2\lambda\mathbf{Y}\mathbf{X}^T$. Strong convexity requires showing that $\langle \nabla f(\mathbf{A}_1) - \nabla f(\mathbf{A}_2), \mathbf{A}_1 - \mathbf{A}_2\rangle \geqslant \mu||\mathbf{A}_1 - \mathbf{A}_2||_F^2$ with $\mu > 0$ for any $\mathbf{A}_1, \mathbf{A}_2$. Using the explicit form of the gradient, strong convexity is equivalent to showing that $\langle \mathbf{X}\mathbf{X}^T + \lambda\text{diag}(\mathbf{X}\mathbf{1}), (\mathbf{A}_1 - \mathbf{A}_2)^T(\mathbf{A}_1 - \mathbf{A}_2)\rangle \geqslant \mu||\mathbf{A}_1 - \mathbf{A}_2||_F^2$. For ease of notation, let $\mathbf{H} = \mathbf{X}\mathbf{X}^T + \lambda\text{diag}(\mathbf{X}\mathbf{1})$ and $\mathbf{G} = (\mathbf{A}_1 - \mathbf{A}_2)^T(\mathbf{A}_1 - \mathbf{A}_2)$. We first note that $\mathbf{H}$ is symmetric positive definite and $\mathbf{G}$ is symmetric positive semidefinite. To see the former claim, it suffices to show that the diagonal entries of diag$(\mathbf{X}\mathbf{1})$ are non-zero. The only case an entry will be zero is if an atom is not used by all data points. For this case, the given atom can be discarded. In all other cases, all the diagonal entries of diag$(\mathbf{X}\mathbf{1})$ are positive. Finally, we claim that $\langle \mathbf{H}\,, \mathbf{G}\rangle \geqslant \lambda_{\min}(\mathbf{H})\,\text{trace}(\mathbf{G})$. This gives the desired strong convexity result with $\mu = \lambda_{\min}(\mathbf{H}) > 0$. Setting the gradient to zero yields the unique solution $\mathbf{A}^* = (1+\lambda)\mathbf{Y}\mathbf{X}^T\mathbf{H}^{-1}$. It remains to prove the claim that $\langle \mathbf{H}\,, \mathbf{G}\rangle \geqslant \lambda_{\min}(\mathbf{H})\,\text{trace}(\mathbf{G})$. This follows from noting that the matrix $\mathbf{H} - \lambda_{\min}(\mathbf{H})\mathbf{I}$ is symmetric positive semidefinite and the term $\langle \mathbf{H} - \lambda_{\min}(\mathbf{H})\mathbf{I}\,, \mathbf{G}\rangle \geqslant 0$ as it is a trace product of symmetric positive semidefinite matrices.

□

Remarks: We note that the weighted ℓ_1 regularizer enables us to obtain strong convexity when optimizing over the dictionary atoms. If $\lambda = 0$, strong convexity is not always guaranteed. We also note that each column of the optimal dictionary is a linear combination of the data points.

D. KDS embedding

In a typical setting, under the manifold hypothesis, the number of landmarks is expected to be much smaller than the number of data points i.e., $m \ll n$. With that, the optimal sparse coefficients obtained from solving (1) are a low-dimensional representation of the high-dimensional data. However when utilizing the sparse coefficients for downstream tasks such as clustering, further dimensionality reduction can be useful. For instance, this will be the case in the setting where $m \ll n$, such that the data is well represented via local landmarks, but $m \gg k$ (e.g., k is number of clusters). In what follows, using connections to spectral clustering and spectral embedding [14], [71], we will show how to obtain low-dimensional embeddings based on the eigenvectors of the covariance matrix $\mathbf{X}\mathbf{X}^T$.

The starting point is the observation that the representation coefficients $\mathbf{X}$ define a bipartite *similarity graph* G with $n+m$ vertices corresponding to the n data points and m learned dictionary atoms. In this graph, each data point $\mathbf{y}_i$ and each atom $\mathbf{a}_j$ is connected by an undirected edge of weight $(\mathbf{x}_i)_j$. To embed the data points and the atoms into $\mathcal{R}^k$, we consider the classic spectral embedding.

$$\min_{\mathbf{Q}\in\mathcal{R}^{k\times(n+m)}} \text{trace}(\mathbf{Q}\mathbf{L}\mathbf{Q}^T) \quad \text{s.t. } \mathbf{Q}\mathbf{Q}^T = \mathbf{I}, \tag{6}$$

where $\mathbf{Q} = [\mathbf{Q}_\mathbf{Y}\ \mathbf{Q}_\mathbf{A}] \in \mathcal{R}^{k\times(n+m)}$. We enforce an additional constraint $\mathbf{Q}_\mathbf{Y} = \mathbf{Q}_\mathbf{A}\mathbf{X}$ to formulate the problem only in terms of the landmarks. We note that this type of assumption has been used for landmark-based locally linear embedding [57]. We will now proceed to state and prove a lemma which shows that the Laplacian quadratic form could be formulated in terms of the landmarks. The Schur complement will be used in the proof and is defined as follows. Consider the block matrix $\mathbf{M} = \begin{pmatrix}\mathbf{A} & \mathbf{B}\\ \mathbf{C} & \mathbf{D}\end{pmatrix}$ where $\mathbf{A} \in \mathcal{R}^{p\times p}$, $\mathbf{B} \in \mathcal{R}^{p\times q}$, $\mathbf{C} \in \mathcal{R}^{q\times p}$ and $\mathbf{D} \in \mathcal{R}^{q\times q}$. If $\mathbf{A}$ is invertible, the Schur complement of $\mathbf{M}$ with respect to $\mathbf{A}$ is defined as $\mathbf{D} - \mathbf{C}\mathbf{A}^{-1}\mathbf{B}$.

Lemma 3. *Let $\mathbf{Q} = [\mathbf{Q}_\mathbf{Y}\ \mathbf{Q}_\mathbf{A}] \in \mathcal{R}^{k\times(n+m)}$. If $\mathbf{Q}_\mathbf{Y} = \mathbf{Q}_\mathbf{A}\mathbf{X}$,*

$$\text{trace}(\mathbf{Q}\mathbf{L}\mathbf{Q}^T) = \text{trace}\left(\mathbf{Q}_\mathbf{A}\mathbf{L}_\mathbf{A}\mathbf{Q}_\mathbf{A}^T\right).$$

Proof. Using the weight matrix in (2), the Laplacian $\mathcal{L}$ is given by $\left[\begin{array}{c|c}\mathbf{I} & -\mathbf{X}^T\\ \hline -\mathbf{X} & \text{diag}(\mathbf{X}\mathbf{1})\end{array}\right]$. We now proceed to evaluate trace$(\mathbf{Q}\mathbf{L}\mathbf{Q}^T)$.

$$\begin{aligned}
&\text{trace}(\mathbf{Q}\mathbf{L}\mathbf{Q}^T) = \text{trace}(\mathbf{Q}^T\mathbf{Q}\mathbf{L})\\
&= \text{trace}\left(\left[\begin{array}{c|c}\mathbf{Q}_\mathbf{Y}^T\mathbf{Q}_\mathbf{Y} & \mathbf{Q}_\mathbf{Y}^T\mathbf{Q}_\mathbf{A}\\ \hline \mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{Y} & \mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{A}\end{array}\right]\left[\begin{array}{c|c}\mathbf{I} & -\mathbf{X}^T\\ \hline -\mathbf{X} & \text{diag}(\mathbf{X}\mathbf{1})\end{array}\right]\right)\\
&=\text{trace}\left((\mathbf{Q}_\mathbf{Y}^T\mathbf{Q}_\mathbf{Y})\mathbf{I} - \mathbf{Q}_\mathbf{Y}^T\mathbf{Q}_\mathbf{A}\mathbf{X} - \mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{Y}\mathbf{X}^T + \mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{A}\mathbf{J}\right)\\
&=\text{trace}\left(\mathbf{X}^T\mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{A}\mathbf{X} - 2\mathbf{X}^T\mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{A}\mathbf{X} + \mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{A}\mathbf{J}\right)\\
&=\text{trace}\left(\mathbf{Q}_\mathbf{A}^T\mathbf{Q}_\mathbf{A}\left(\mathbf{J} - \mathbf{X}\mathbf{X}^T\right)\right) = \text{trace}\left(\mathbf{Q}_\mathbf{A}\mathbf{L}_\mathbf{A}\mathbf{Q}_\mathbf{A}^T\right),
\end{aligned}$$

where $\mathbf{J} = \text{diag}(\mathbf{X}\mathbf{1})$ and $\mathbf{L}_\mathbf{A}$ is known as the *Schur complement* of $\mathbf{L}$ with respect to $\mathbf{Y}$. □

Given the above proof, we consider the following spectral embedding problem

$$\min_{\mathbf{Q}_\mathbf{A}\in\mathcal{R}^{k\times m}} \text{trace}(\mathbf{Q}_\mathbf{A}\mathbf{L}_\mathbf{A}\mathbf{Q}_\mathbf{A}^T) \quad \text{s.t. } \mathbf{Q}_\mathbf{A}\mathbf{Q}_\mathbf{A}^T = \mathbf{I}. \tag{7}$$

The above problem is a standard spectral problem whose optimal solution is $\mathbf{Q}_\mathbf{A}^* = \mathbf{U}_k^T$ where the columns of $\mathbf{U}_k \in \mathcal{R}^{m\times k}$ are the eigenvectors of $\mathbf{L}_\mathbf{A}$ corresponding to the largest k eigenvalues. It follows that the dominant computation of the KDS spectral embedding only requires the calculation of the first k eigenvectors of an $m \times m$ matrix $\mathbf{L}_\mathbf{A}$, which is very small when $m \ll n$, as well as a handful of $O(mn)$-time multiplications by the matrix $\mathbf{X}$ to compute the adjacency matrix $\mathbf{X}\mathbf{X}^\top$ and recover $\mathbf{Q}_\mathbf{Y} = \mathbf{Q}_\mathbf{A}\mathbf{X}$.

We note that it is important to set k and m carefully. In lack of prior knowledge about number of clusters, one could employ the eigengap heuristic [72] which sets number of clusters based on the gap between eigenvalues of the graph Laplacian. In terms of m, a relatively large value of m, implies that points are well represented via local landmarks. However, this has the implication that points within the same cluster may not have the same sparsity structure. In contrast, a relatively small value of m would allow points from different clusters to have a similar sparsity structure (which leads to sub-optimal clustering).

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

V. Dictionary learning algorithm

In this section, given a set of data points, we discuss the problem of estimating both the sparse representations and dictionary atoms. To this end, we study the following minimization problem:

$$\min_{\mathbf{A}\in\mathcal{R}^{d\times m},\mathbf{X}\in S} \sum_{i=1}^{n}\left[\frac{1}{2}\|\mathbf{y}_i-\mathbf{A}\mathbf{x}_i\|^2+\lambda\sum_{j=1}^{m}(\mathbf{x}_i)_j\|\mathbf{y}_i-\mathbf{a}_j\|^2\right], \tag{8}$$

where $\mathbf{X}=[\mathbf{x}_1,\mathbf{x}_2,...,\mathbf{x}_n]$ with each $\mathbf{x}_i\in\Delta^m$. The balance between the reconstruction loss and the locality regularization is controlled by the parameter λ. A standard way to solve the above minimization program is alternating minimization which alternates between sparse approximation and dictionary update steps [73]. We discuss the two steps below. The KDS algorithm is summarized in in Algorithm 1.

A. Sparse coding

Given a fixed dictionary $\mathbf{A}$, the sub-problem over the sparse coefficients is a weighted ℓ_1 minimization problem for which efficient methods exist [74]. We consider the accelerated projected gradient descent algorithm [75] to solve this problem. Since the minimization problem for $\mathbf{X}$ decouples into optimizing the sparse representation of each data point, we consider the problem of finding the optimal coefficient given the dictionary $\mathbf{A}$ and a data point $\mathbf{y}$ as follows

$$\mathbf{x}^*(\mathbf{A},\mathbf{y})=\operatorname*{argmin}_{\mathbf{x}\in\Delta^m}\ \frac{1}{2}\|\mathbf{y}-\mathbf{A}\mathbf{x}\|^2+\lambda\sum_{j=1}^{m}x_j\|\mathbf{y}-\mathbf{a}_j\|^2. \tag{9}$$

Let $\mathcal{L}(\mathbf{A},\mathbf{y},\mathbf{x},\lambda)$ denote the objective in the above program.

The accelerated projected gradient descent: This method starts with the initialization $\mathbf{x}^0=\tilde{\mathbf{x}}^{(0)}=\mathbf{0}$ and considers the following updates

$$\mathbf{x}^{(t+1)}=\mathcal{P}_{\Delta^m}\left(\tilde{\mathbf{x}}^{(t)}-\alpha\nabla_{\mathbf{x}}\mathcal{L}(\mathbf{A},\mathbf{y},\tilde{\mathbf{x}}^{(t)})\right)$$
$$\tilde{\mathbf{x}}^{(t+1)}=\mathbf{x}^{(t+1)}+\frac{t-1}{t+2}(\mathbf{x}^{(t+1)}-\mathbf{x}^{(t)}).$$

for $0\leqslant t\leqslant T_{\max}$. The operator $\mathcal{P}_{\Delta^m}$ projects onto S, the probability simplex and has a closed form that can be readily computed [76], [77]. The parameter α is a step size. We note below the gradient of $\mathcal{L}$ with respect to $\mathbf{x}_i$:

$$\nabla_{\mathbf{x}_i}\mathcal{L}(\mathbf{A},\mathbf{y}_i,\mathbf{x}_i,\lambda)=\mathbf{A}^\top(\mathbf{A}\mathbf{x}_i-\mathbf{y}_i)+\lambda\sum_{j=1}^{m}\|\mathbf{y}_i-\mathbf{a}_j\|^2\mathbf{e}_j$$

B. Dictionary learning

After $T_{\max}$ iterations of the sparse coding step, we have optimized sparse coefficients $\{\mathbf{x}_i^{(T_{\max})}\}_{i=1}^n$ corresponding to the data points $\{\mathbf{y}_i\}_{i=1}^n$. The next part of the algorithm is to optimize for the dictionary which can be estimated by solving the following optimization problem:

$$\min_{\mathbf{A}\in\mathcal{R}^{d\times m}} \sum_{i=1}^{n}\left[\frac{1}{2}\|\mathbf{y}_i-\mathbf{A}\mathbf{x}_i^{(T_{\max})}\|^2+\lambda\sum_{j=1}^{m}(\mathbf{x}_i^{(T_{\max})})_j\|\mathbf{y}_i-\mathbf{a}_j\|^2\right]. \tag{10}$$

Let $\mathcal{L}_1(\mathbf{A},\mathbf{y},\mathbf{x},\lambda)$ denote the objective in the above program. We note that the gradient of $\mathcal{L}_1$ with respect to $\mathbf{A}$ is given by

$$\nabla_{\mathbf{A}}\mathcal{L}_1=2(\mathbf{A}\mathbf{X}-\mathbf{Y})\mathbf{X}^T+2\lambda\mathbf{A}\mathrm{diag}(\mathbf{X}\mathbf{1})-2\lambda\mathbf{Y}\mathbf{X}^T$$

The dictionary learning sub-problem can be solved using gradient descent.

C. Complexity of alternating minimization

For a fixed data point, the gradient update to estimate the coefficient is $O(md)$ and the projection onto the simplex is $O(m\log(m))$ [76]. Therefore, the per-iteration cost of sparse coding is $O(nm\max(\log(m),d))$. The per-iteration complexity of the dictionary learning step is $O(nmd)$ which is the cost of the gradient update.

Algorithm 1 KDS algorithm to solve (8)

```
Input: Data points $\mathbf{Y}=[\mathbf{y}_1,\ldots,\mathbf{y}_n]\in\mathcal{R}^{d\times n}$, maxiterations.
Initialization: $\mathbf{x}_i^{(0)}=\mathbf{0}$ for $1\leqslant i\leqslant n$. Set $\mathbf{A}^{(0)}\in\mathcal{R}^{d\times m}$ to be random subset of data.
for k = 1:maxiterations do
    Set step size: $\alpha=\frac{1}{(\sigma_{\max}(\mathbf{A}^{(k-1)}))^2}$.
    Sparse coding via encoder: Given $\mathbf{A}^{(k-1)}$, use accelerated project gradient descent to obtain $\{\mathbf{x}_1^{(k)},\mathbf{x}_2^{(k)},...,\mathbf{x}_n^{(k)}\}$.
    Decoder: Reconstruct approximate data $\{\mathbf{A}^{(k-1)}\mathbf{x}_1^{(k)},...,\mathbf{A}^{(k-1)}\mathbf{x}_n^{(k)}\}$.
    Dictionary learning: Backpropagation to obtain $\mathbf{A}^{(k)}$.
```

D. Algorithm unrolling

In order to solve (8) efficiently and to design an interpretable network, we consider a technique known as algorithm unrolling. This is the process of designing a highly-structured recurrent neural network to efficiently solve problems [78]. Although our application of the technique for manifold learning is new, there exists a rich literature on the subject in the context of sparse dictionary learning [18]–[20], [22]–[25]. In order to solve the relaxed optimization problem in (10), we introduce an autoencoder architecture that implicitly solves the problem when trained by backpropagation. Given a dictionary $\mathbf{A}$, our encoder maps a data point $\mathbf{y}$, or a batch of such points, to the sparse code $\mathbf{x}$ minimizing $\mathcal{L}(\mathbf{A},\mathbf{y},\mathbf{x})$. This is done by unfolding T iterations of projected gradient descent on $\mathcal{L}$ into a deep recurrent neural network. Our linear decoder reconstructs the input as $\hat{\mathbf{y}}=\mathbf{A}\mathbf{x}$. The network weights correspond to the dictionary $\mathbf{A}$, which is initialized to a random subset of the data $\mathbf{Y}$ and then trained to minimize (8) by backpropagation through the entire autoencoder. If we view the forward pass through our encoder as an analogue of the sparse recovery step used in traditional alternating-minimization schemes, then this backward pass corresponds to an enhanced version of the so-called "dictionary update" step. We note that the projection onto the probability simplex can be written as a modified ReLU function and thus serves as a non-linear activation function in the encoder.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

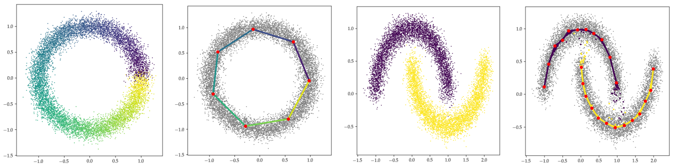

Fig. 3. Circle and two moons. Autoencoder input (first and third) and output (second and fourth), with learned atoms marked in red.

VI. Experiments

A. Application of KDS to clustering

Let $\mathbf{Y} = [\mathbf{y}_1, \ldots, \mathbf{y}_n] \in \mathcal{R}^{d \times n}$ be a collection of n data points in $\mathcal{R}^d$. To cluster the data, we utilize Algorithm 1 to obtain sparse representation coefficients $\mathbf{X}$ and a set of m atoms $\mathbf{a}_1, \ldots, \mathbf{a}_m$. Our similarity matrix is $\mathbf{X}\mathbf{X}^T$. Given the similarity matrix, to cluster the data into k clusters, we apply spectral clustering which first embeds the data using k eigenvectors of a normalized graph Laplacian corresponding to the largest k eigenvalues. We note that the obtained embedding is extended to all data points by applying the dictionary. The details of these are in Section IV. D. Given the embedding, we run k-means to obtain the cluster labels [71].

In this section, we demonstrate the ability of KDS, implemented in PyTorch [79], to efficiently and accurately recover the underlying clusters of both synthetic and real-world data sets. Details about pre-processing of data and parameter selection for KDS as well as baseline algorithms can be found in the Supplementary Materials. All clustering experiments are evaluated with respect to a given ground truth clustering using the unsupervised clustering accuracy (ACC), which is invariant under a permutation of the cluster labels. Accuracy is defined as the percentage of correct matches with respect to the ground truth labels of the data.

B. Synthetic Data

Learned Dictionary Atoms: For our first experiment, we visualize the dictionary atoms learned by our autoencoder when the data is sampled from one-dimensional manifolds in $\mathcal{R}^2$. Figure 3 shows two such data sets.

The first is the unit circle in $\mathcal{R}^2$. The second is the classic two moon data set [71], which consists of two disjoint semicircular arcs in $\mathcal{R}^2$. For each of these two data sets, we trained the autoencoder on data sampled uniformly from the underlying manifold(s). We added small Gaussian white noise to each data point to make the representation learning problem more challenging. Figure 3 shows the result of training the autoencoder on these data sets. We see that in each case, the atoms learned by the model are meaningful. Moreover, in each case, we accurately reconstruct each data point as sparse convex combinations of these atoms, up to the additive white noise. As a final remark, drawing a sample of 5000 data points from the noisy two moons distribution, computing their sparse coefficients, and performing spectral clustering with $k = 2$ on the associated bipartite similarity graph results in a clustering accuracy of 99.9%. We note that KDS outperforms baseline algorithms (see Table II).

Clustering with Narrow Separation: Our next experiment assesses the clustering capabilities of our algorithm in a toy setting. We studied a simple family of data distributions consisting of two underlying clusters in $\mathcal{R}^2$. These clusters took the shape of two concentric circles of radii $r_{\text{outer}} = 1$ and $r_{\text{inner}} = 1 - \delta$, where $\delta \in [0, 1]$ is a separation parameter. For multiple values of δ, we trained our structured autoencoder with m atoms on data sampled uniformly from these two manifolds, each with half the probability mass. For this experiment, we did not add any Gaussian noise to the data.

Figure 4 shows the results across a range of m and δ. Figure 5 shows the accuracy achieved by performing spectral clustering on the corresponding similarity graphs. Based on these results, it appears that our clustering algorithm is capable of distinguishing between clusters of arbitrarily small separation δ, provided that the number of atoms is sufficiently large.

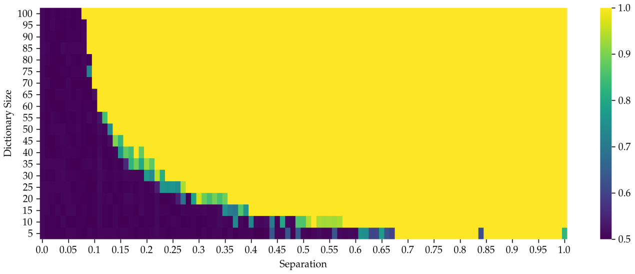

Fig. 4. Clustering accuracy for concentric circles across δ, m.

C. Real-World Data

In this section, we empirically evaluate our algorithm on synthetic and three publicly available real-world data sets. We compared our method against four baseline clustering algorithms that may be interpreted as dictionary learning: (i) k-means (KM) [80], which learns a single dictionary atom for each cluster; (ii) SMCE, which solves a sparse optimization problem over a global dictionary consisting of all data points, then runs spectral clustering on a similarity graph derived from the solution [41]; (iii) LLL [57] which is a landmark method that uses uniform sampling (LLL-U) or k-means clustering (LLL-K) and (iv) ESC [59] is a landmark method that uses furthest first search. A summary of results can be found in Table II. For LLL, clustering is based on an affinity matrix built from the weights. The reported accuracy is the best result after optimizing for different factors (# of neighbors, exemplar scheme, optimal clustering). See the Supplementary Materials for details of numerical experiments. We note that the linear system utilized in LLL is ill-conditioned, due to few sample points to characterize the manifold, for the Yale B dataset and obtains poor results. We denote this result by NA. Similarly, clustering on MNIST is ill-conditioned with 500 points and we instead set $m = 800$.

MNIST Handwritten Digit Database: The database [81] consists of 28×28 grayscale images of 10 different digits. We ran our clustering on a subset of the data comprised of the $k = 5$ digits $\{0, 3, 4, 6, 7\}$, following the example of [41]. Figure 1 shows a subset of the randomly initialized atoms for

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

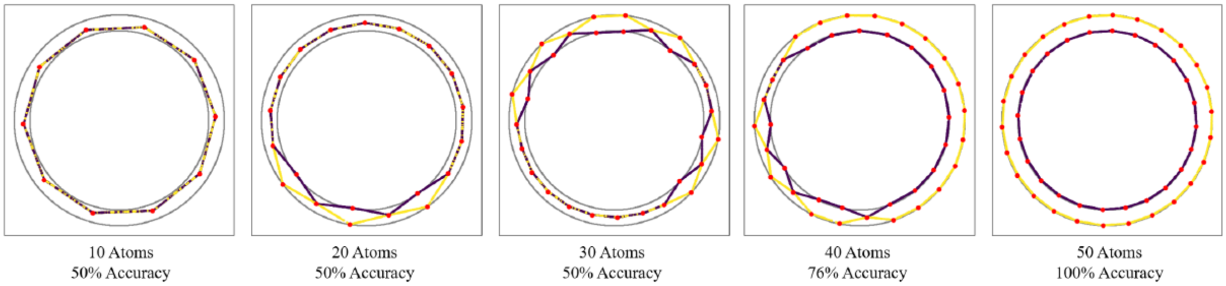

Fig. 5. (a) Autoencoder output and learned atoms for concentric circles, separation $\delta = 0.15$.

TABLE II
CLUSTERING ACCURACIES FOR VARIOUS DATA SETS ROUNDED TO THREE DIGITS.

Method	Moons	MNIST-5	Yale B	Salinas-A
KM	0.756	0.887	0.508	0.774
SMCE	0.835	0.975	1.0	0.847
KDS	**0.999**	**0.986**	**1.0**	**0.881**
LLL-U	0.944	0.976	NA	0.285
LLL-K	0.950	0.980	NA	0.261
ESC	0.842	0.966	0.958	0.840

MNIST before training (black and white) and after training and clustering (color).

Extended Yale Face Database B: The cropped version of the database [82] consists of 192 × 168 grayscale images of 39 different faces under varying illumination conditions. We ran our algorithm on a subset of the data comprised of $k = 2$ subjects.

Salinas-A Hyperspectral Image: The Salinas-A data set is a single aerial-view hyperpspectral image of the Salinas valley in California with 224 bands and 6 regions corresponding to different crops [83]. We ran our algorithm on the entire 86×83 pixel image with $k = 6$ segments. Regarding the KDS result depicted in Figure 6, KDS exhibits a specific limitation: it tends to blend certain elements of the aquamarine class with the yellow class, a characteristic shared with many hyperspectral image (HSI) clustering algorithms. Conversely, K-means exhibits a distinct challenge as it fails not only to distinguish the turquoise class but also struggles to accurately separate a portion of the aquamarine class.

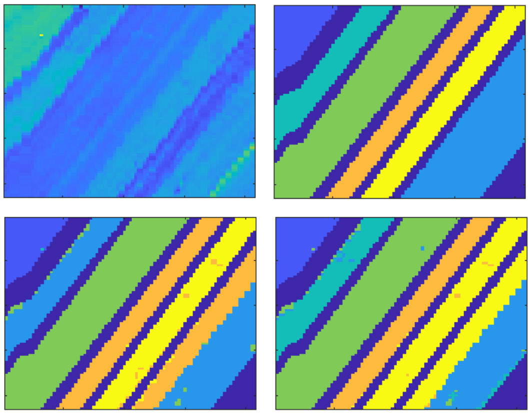

Fig. 6. Salinas-A Scene. From left to right and top to bottom: image data (mean across spectral bands), ground truth clusters, predicted clusters by K-means, predicted clusters by KDS.

VII. Conclusion

In this paper, we proposed a structured dictionary learning algorithm K-Deep Simplex (KDS) that combines nonlinear dimensionality reduction and sparse coding. Given a set of data points as an input, KDS learns a dictionary along with sparse coefficients supported on the probability simplex. Assuming that data points are generated from a convex combination of atoms, represented as vertices of a unique Delaunay triangulation, we prove that the proposed regularization recovers the underlying sparse solution. Furthermore, we demonstrate that when a data point undergoes perturbation and the perturbed point resides within the same d-simplex as the original point, we establish the stability of sparse representations. We also show how the optimization problem for KDS can be recast and solved via a structured deep autoencoder. We then discuss how KDS can be applied for the clustering problem by constructing a similarity graph based on the obtained representation coefficients. Our experiments show that KDS learns meaningful representation and obtains competitive results while offering dramatic savings in running time. In contrast to methods that set the dictionary to be the set of all data points, KDS is quasilinear with the number of dictionary atoms and offers a scalable framework. In our future work, we intend to explore several aspects, including stability estimates for scenarios where perturbed and original data points are located in adjacent d-simplices, conducting experiments on large, real-world datasets, examining the sampling of data manifold using KDS and drawing comparisons to [51], [52], investigating the out-of-sample extension property of KDS, and exploring the generative capabilities of the model.

Acknowledgements: AT acknowledges support from NSF through grant DMS-2208392. JMM gratefully acknowledges support from the NSF through grants DMS-1912737, DMS-1924513, DMS-2309519, and DMS-2318894 as well as The Camille & Henry Dreyfus Foundation. DB acknowledges support from the NSF through grants DMS-2134157 and PHY-2019786.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

References

[1] C. Loader, *Local regression and likelihood*. Springer Science & Business Media, 2006.

[2] C. J. Stone, "Consistent nonparametric regression," *The annals of statistics*, pp. 595–620, 1977.

[3] W. S. Cleveland, "Robust locally weighted regression and smoothing scatterplots," *Journal of the American statistical association*, vol. 74, no. 368, pp. 829–836, 1979.

[4] D. H. McLain, "Drawing contours from arbitrary data points," *The Computer Journal*, vol. 17, no. 4, pp. 318–324, 1974.

[5] J. H. Friedman and W. Stuetzle, "Projection pursuit regression," *Journal of the American statistical Association*, vol. 76, no. 376, pp. 817–823, 1981.

[6] B. Li, J. Friedman, R. Olshen, and C. Stone, "Classification and regression trees (cart)," *Biometrics*, vol. 40, no. 3, pp. 358–361, 1984.

[7] D. Hallac, J. Leskovec, and S. Boyd, "Network lasso: Clustering and optimization in large graphs," in *Proceedings of the 21th ACM SIGKDD international conference on knowledge discovery and data mining*, 2015, pp. 387–396.

[8] M. Yamada, T. Koh, T. Iwata, J. Shawe-Taylor, and S. Kaski, "Localized lasso for high-dimensional regression," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 325–333.

[9] M. Petrovich and M. Yamada, "Fast local linear regression with anchor regularization," *arXiv preprint arXiv:2003.05747*, 2020.

[10] J. M. Lee, "Smooth manifolds," in *Introduction to Smooth Manifolds*. Springer, 2013, pp. 1–31.

[11] B. Schölkopf, A. Smola, and K.-R. Müller, "Kernel principal component analysis," in *International conference on artificial neural networks*. Springer, 1997, pp. 583–588.

[12] J. B. Tenenbaum, V. De Silva, and J. C. Langford, "A global geometric framework for nonlinear dimensionality reduction," *science*, vol. 290, no. 5500, pp. 2319–2323, 2000.

[13] S. T. Roweis and L. K. Saul, "Nonlinear dimensionality reduction by locally linear embedding," *science*, vol. 290, no. 5500, pp. 2323–2326, 2000.

[14] M. Belkin and P. Niyogi, "Laplacian eigenmaps for dimensionality reduction and data representation," *Neural computation*, vol. 15, no. 6, pp. 1373–1396, 2003.

[15] R. R. Coifman and S. Lafon, "Diffusion maps," *Applied and computational harmonic analysis*, vol. 21, no. 1, pp. 5–30, 2006.

[16] A. Cutler and L. Breiman, "Archetypal analysis," *Technometrics*, vol. 36, no. 4, pp. 338–347, 1994.

[17] D. van Dijk, D. B. Burkhardt, M. Amodio, A. Tong, G. Wolf, and S. Krishnaswamy, "Finding archetypal spaces using neural networks," in *2019 IEEE International Conference on Big Data (Big Data)*. IEEE, 2019, pp. 2634–2643.

[18] B. Tolooshams, S. Dey, and D. Ba, "Deep residual autoencoders for expectation maximization-inspired dictionary learning," *IEEE Transactions on Neural Networks and Learning Systems*, 2020.

[19] B. Tolooshams, A. Song, S. Temereanca, and D. Ba, "Convolutional dictionary learning based auto-encoders for natural exponential-family distributions," in *International Conference on Machine Learning*. PMLR, 2020, pp. 9493–9503.

[20] B. Tolooshams and D. Ba, "Stable and interpretable unrolled dictionary learning," *Transactions on Machine Learning Research*, 2022.

[21] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm unrolling: Interpretable, efficient deep learning for signal and image processing," *IEEE Signal Processing Magazine*, vol. 38, no. 2, pp. 18–44, 2021.

[22] T. Chang, B. Tolooshams, and D. Ba, "Randnet: deep learning with compressed measurements of images," in *2019 IEEE 29th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2019, pp. 1–6.

[23] K. Gregor and Y. LeCun, "Learning fast approximations of sparse coding," in *Proceedings of the 27th international conference on international conference on machine learning*, 2010, pp. 399–406.

[24] J. T. Rolfe and Y. LeCun, "Discriminative recurrent sparse auto-encoders: 1st international conference on learning representations, iclr 2013," in *1st International Conference on Learning Representations, ICLR 2013*, 2013.

[25] B. Tolooshams, S. Dey, and D. Ba, "Scalable convolutional dictionary learning with constrained recurrent sparse auto-encoders," in *2018 IEEE 28th International Workshop on Machine Learning for Signal Processing (MLSP)*. IEEE, 2018, pp. 1–6.

[26] A. Tasissa, P. Tankala, and D. Ba, "Weighed l1 on the simplex: Compressive sensing meets locality," in *2021 IEEE Statistical Signal Processing Workshop (SSP)*, 2021, pp. 476–480.

[27] P. Tankala, A. Tasissa, J. M. Murphy, and D. Ba, "K-deep simplex: Deep manifold learning via local dictionaries," *arXiv preprint arXiv:2012.02134*, 2020.

[28] T. Cour, P. Srinivasan, and J. Shi, "Balanced graph matching," *Advances in neural information processing systems*, vol. 19, 2006.

[29] Y. Chen, D. M. Paiton, and B. A. Olshausen, "The sparse manifold transform," in *Proceedings of the 32nd International Conference on Neural Information Processing Systems*, 2018, pp. 10 534–10 545.

[30] B. A. Olshausen and D. J. Field, "Emergence of simple-cell receptive field properties by learning a sparse code for natural images," *Nature*, vol. 381, no. 6583, pp. 607–609, 1996.

[31] A. M. Bruckstein, D. L. Donoho, and M. Elad, "From sparse solutions of systems of equations to sparse modeling of signals and images," *SIAM review*, vol. 51, no. 1, pp. 34–81, 2009.

[32] K. Engan, S. O. Aase, and J. H. Husøy, "Multi-frame compression: Theory and design," *Signal Processing*, vol. 80, no. 10, pp. 2121–2140, 2000.

[33] M. Aharon, M. Elad, and A. Bruckstein, "K-svd: An algorithm for designing overcomplete dictionaries for sparse representation," *IEEE Transactions on signal processing*, vol. 54, no. 11, pp. 4311–4322, 2006.

[34] W. K. Allard, G. Chen, and M. Maggioni, "Multi-scale geometric methods for data sets ii: Geometric multi-resolution analysis," *Applied and computational harmonic analysis*, vol. 32, no. 3, pp. 435–462, 2012.

[35] M. Maggioni, S. Minsker, and N. Strawn, "Multiscale dictionary learning: non-asymptotic bounds and robustness," *The Journal of Machine Learning Research*, vol. 17, no. 1, pp. 43–93, 2016.

[36] F. Dornaika and L. Weng, "Sparse graphs with smoothness constraints: Application to dimensionality reduction and semi-supervised classification," *Pattern Recognition*, vol. 95, pp. 285–295, 2019.

[37] D. Cai, X. He, J. Han, and T. S. Huang, "Graph regularized nonnegative matrix factorization for data representation," *IEEE transactions on pattern analysis and machine intelligence*, vol. 33, no. 8, pp. 1548–1560, 2010.

[38] H. Hu, Z. Lin, J. Feng, and J. Zhou, "Smooth representation clustering," in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 3834–3841.

[39] K. Yu, T. Zhang, and Y. Gong, "Nonlinear learning using local coordinate coding," *Advances in neural information processing systems*, vol. 22, 2009.

[40] J. Wang, J. Yang, K. Yu, F. Lv, T. Huang, and Y. Gong, "Locality-constrained linear coding for image classification," in *2010 IEEE computer society conference on computer vision and pattern recognition*. IEEE, 2010, pp. 3360–3367.

[41] E. Elhamifar and R. Vidal, "Sparse manifold clustering and embedding," *Advances in neural information processing systems*, vol. 24, pp. 55–63, 2011.

[42] T. Ding, S. Tong, K. H. R. Chan, X. Dai, Y. Ma, and B. D. Haeffele, "Unsupervised manifold linearizing and clustering," *arXiv preprint arXiv:2301.01805*, 2023.

[43] G. Zhong and C.-M. Pun, "Subspace clustering by simultaneously feature selection and similarity learning," *Knowledge-Based Systems*, vol. 193, p. 105512, 2020.

[44] M. Abdolali and N. Gillis, "Beyond linear subspace clustering: A comparative study of nonlinear manifold clustering algorithms," *Computer Science Review*, vol. 42, p. 100435, 2021.

[45] M. Zheng, J. Bu, C. Chen, C. Wang, L. Zhang, G. Qiu, and D. Cai, "Graph regularized sparse coding for image representation," *IEEE transactions on image processing*, vol. 20, no. 5, pp. 1327–1336, 2010.

[46] J. Huang, F. Nie, and H. Huang, "A new simplex sparse learning model to measure data similarity for clustering," in *Twenty-fourth international joint conference on artificial intelligence*, 2015.

[47] Y. Zhou and K. E. Barner, "Locality constrained dictionary learning for nonlinear dimensionality reduction," *IEEE Signal Processing Letters*, vol. 20, no. 4, pp. 335–338, 2013.

[48] K. Jiang, Z. Liu, Z. Liu, and Q. Sun, "Locality constrained analysis dictionary learning via k-svd algorithm," *arXiv preprint arXiv:2104.14130*, 2021.

[49] H.-F. Yin, X.-J. Wu, and S.-G. Chen, "Locality constraint dictionary learning with support vector for pattern classification," *IEEE Access*, vol. 7, pp. 175 071–175 082, 2019.

[50] W. Liao, M. Maggioni, and S. Vigogna, "Multiscale regression on unknown manifolds," *Mathematics in Engineering*, vol. 4, no. 4, pp. 1–25, 2022. [Online]. Available: https://www.aimspress.com/article/doi/10.3934/mine.2022028

[51] V. Silva and J. Tenenbaum, "Global versus local methods in nonlinear dimensionality reduction," *Advances in neural information processing systems*, vol. 15, pp. 721–728, 2002.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.

This article has been accepted for publication in IEEE Transactions on Signal Processing. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TSP.2023.3322820

[52] V. De Silva and J. B. Tenenbaum, "Sparse multidimensional scaling using landmark points," technical report, Stanford University, Tech. Rep., 2004.

[53] W. S. Torgerson, "Multidimensional scaling: I. theory and method," *Psychometrika*, vol. 17, no. 4, pp. 401–419, 1952.

[54] J. C. Gower, "Some distance properties of latent root and vector methods used in multivariate analysis," *Biometrika*, vol. 53, no. 3-4, pp. 325–338, 1966.

[55] G. Young and A. S. Householder, "Discussion of a set of points in terms of their mutual distances," *Psychometrika*, vol. 3, no. 1, pp. 19–22, 1938.

[56] X. Chen and D. Cai, "Large scale spectral clustering with landmark-based representation," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 25, no. 1, 2011, pp. 313–318.

[57] M. Vladymyrov and M. Á. Carreira-Perpinán, "Locally linear landmarks for large-scale manifold learning," in *Joint European Conference on Machine Learning and Knowledge Discovery in Databases*. Springer, 2013, pp. 256–271.

[58] W. Liu, J. He, and S.-F. Chang, "Large graph construction for scalable semi-supervised learning," in *Proceedings of the 27th International Conference on International Conference on Machine Learning*. Madison, WI, USA: Omnipress, 2010, p. 679–686.

[59] C. You, C. Li, D. P. Robinson, and R. Vidal, "A scalable exemplar-based subspace clustering algorithm for class-imbalanced data," in *European Conference on Computer Vision*. Springer, 2018, pp. 68–85.

[60] M. Abdolali, N. Gillis, and M. Rahmati, "Scalable and robust sparse subspace clustering using randomized clustering and multilayer graphs," *Signal Processing*, vol. 163, pp. 166–180, 2019.

[61] S. Matsushima and M. Brbic, "Selective sampling-based scalable sparse subspace clustering," *Advances in Neural Information Processing Systems*, vol. 32, 2019.

[62] D. D. Lee and H. S. Seung, "Learning the parts of objects by non-negative matrix factorization," *Nature*, vol. 401, no. 6755, pp. 788–791, 1999.

[63] N. Gillis, *Nonnegative Matrix Factorization*, ser. Data Science. Society for Industrial and Applied Mathematics, 2020. [Online]. Available: https://books.google.com/books?id=6dUPEAAAQBAJ

[64] C. H. Ding, T. Li, and M. I. Jordan, "Convex and semi-nonnegative matrix factorizations," *IEEE transactions on pattern analysis and machine intelligence*, vol. 32, no. 1, pp. 45–55, 2008.

[65] C.-H. Lin, R. Wu, W.-K. Ma, C.-Y. Chi, and Y. Wang, "Maximum volume inscribed ellipsoid: A new simplex-structured matrix factorization framework via facet enumeration and convex optimization," *SIAM Journal on Imaging Sciences*, vol. 11, no. 2, pp. 1651–1679, 2018.

[66] M. Abdolali and N. Gillis, "Simplex-structured matrix factorization: Sparsity-based identifiability and provably correct algorithms," *SIAM Journal on Mathematics of Data Science*, vol. 3, no. 2, pp. 593–623, 2021.

[67] J. B. Greer, "Sparse demixing of hyperspectral images," *IEEE Transactions on image processing*, vol. 21, no. 1, pp. 219–228, 2011.

[68] A. S. Charles, B. A. Olshausen, and C. J. Rozell, "Learning sparse codes for hyperspectral imagery," *IEEE Journal of Selected Topics in Signal Processing*, vol. 5, no. 5, pp. 963–978, 2011.

[69] P. Cignoni, C. Montani, and R. Scopigno, "Dewall: A fast divide and conquer delaunay triangulation algorithm in ed," *Computer-Aided Design*, vol. 30, no. 5, pp. 333–341, 1998.

[70] L. Chen and J.-c. Xu, "Optimal delaunay triangulations," *Journal of Computational Mathematics*, pp. 299–308, 2004.

[71] A. Ng, M. Jordan, and Y. Weiss, "On spectral clustering: Analysis and an algorithm," *Advances in neural information processing systems*, vol. 14, pp. 849–856, 2001.

[72] U. Von Luxburg, "A tutorial on spectral clustering," *Statistics and computing*, vol. 17, pp. 395–416, 2007.

[73] A. Agarwal, A. Anandkumar, P. Jain, and P. Netrapalli, "Learning sparsely used overcomplete dictionaries via alternating minimization," *SIAM Journal on Optimization*, vol. 26, no. 4, pp. 2775–2799, 2016.

[74] M. Salman Asif and J. Romberg, "Fast and Accurate Algorithms for Re-Weighted L1-Norm Minimization," *arXiv e-prints*, 2012.

[75] W. Su, S. Boyd, and E. Candes, "A differential equation for modeling nesterov's accelerated gradient method: theory and insights," *Advances in neural information processing systems*, vol. 27, 2014.

[76] W. Wang and M. A. Carreira-Perpinán, "Projection onto the probability simplex: An efficient algorithm with a simple proof, and an application," *arXiv preprint arXiv:1309.1541*, 2013.

[77] L. Condat, "Fast projection onto the simplex and the l1 ball," *Mathematical Programming*, vol. 158, no. 1, pp. 575–585, 2016.

[78] V. Monga, Y. Li, and Y. C. Eldar, "Algorithm Unrolling: Interpretable, Efficient Deep Learning for Signal and Image Processing," *arXiv e-prints*, 2019.

[79] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, vol. 32, pp. 8026–8037, 2019.

[80] S. Lloyd, "Least squares quantization in pcm," *IEEE transactions on information theory*, vol. 28, no. 2, pp. 129–137, 1982.

[81] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proceedings of the IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[82] K.-C. Lee, J. Ho, and D. J. Kriegman, "Acquiring linear subspaces for face recognition under variable lighting," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 27, no. 5, pp. 684–698, 2005.

[83] B. A. M Graña, MA Veganzons, "Hyperspectral remote sensing scenes," http://www.ehu.eus/ccwintco/index.php/Hyperspectral_Remote_Sensing_Scenes, accessed: 2020-02-04.

© 2023 IEEE. Personal use is permitted, but republication/redistribution requires IEEE permission. See https://www.ieee.org/publications/rights/index.html for more information.