Pattern Recognition Letters 32 (2011) 1299–1304

Contents lists available at ScienceDirect

Pattern Recognition Letters

journal homepage: www.elsevier.com/locate/patrec

Optimality and stability of the *K*-hyperline clustering algorithm

Jayaraman J. Thiagarajan, Karthikeyan N. Ramamurthy *, Andreas Spanias

SenSIP Center, School of ECEE, Arizona State University, AZ 85287-5706, USA

ARTICLE INFO

Article history:
Received 23 June 2010
Available online 12 March 2011
Communicated by P. Franti

Keywords:
K-hyperline clustering
Optimality
Stability
Voronoi
Empirical risk minimization

ABSTRACT

K-hyperline clustering is an iterative algorithm based on singular value decomposition and it has been successfully used in sparse component analysis. In this paper, we prove that the algorithm converges to a locally optimal solution for a given set of training data, based on Lloyd's optimality conditions. Furthermore, the local optimality is shown by developing an Expectation-Maximization procedure for learning dictionaries to be used in sparse representations and by deriving the clustering algorithm as its special case. The cluster centroids obtained from the algorithm are proved to tessellate the space into convex Voronoi regions. The stability of clustering is shown by posing the problem as an empirical risk minimization procedure over a function class. It is proved that, under certain conditions, the cluster centroids learned from two sets of i.i.d. training samples drawn from the same probability space become arbitrarily close to each other, as the number of training samples increase asymptotically.

1. Introduction

The *K*-hyperline clustering algorithm is an iterative *K*-means like procedure that performs a least squares fit of *K* 1-D linear subspaces to the training data (He et al., 2009). These subspaces are referred to as hyperlines. Both the *K*-means and *K*-hyperline clustering algorithms are special cases of the general joint optimization problem of sparse representation and dictionary learning. The general model for representation that we consider here is

$$\mathbf{Y} = \mathbf{\Psi}\mathbf{A} + \mathbf{N}, \tag{1}$$

where the matrix $\mathbf{Y} = [\mathbf{y}_1\mathbf{y}_2, \ldots, \mathbf{y}_T] \in \mathbb{R}^{M\times T}$ is a collection of T training vectors and each training vector $\mathbf{y}_i \in \mathbb{R}^M$. $\mathbf{\Psi} = [\psi_1\psi_2, \ldots, \psi_K] \in \mathbb{R}^{M\times K}$ is a dictionary that contains the set of representative patterns. $\mathbf{A} = [\mathbf{a}_1\mathbf{a}_2, \ldots, \mathbf{a}_T] \in \mathbb{R}^{K\times T}$ is the matrix of T coefficient vectors and $\mathbf{N}$ is a noise matrix whose elements are independent realizations from the Gaussian distribution $\mathcal{N}(0, \sigma^2)$. We will assume that the column vectors of $\mathbf{\Psi}$ are normalized. The dictionary $\mathbf{\Psi}$ can consist of a predefined set of patterns based on the mathematical model of data or alternatively they can be learned from the training vectors themselves. Learning the dictionary from data leads to superior performance in applications such as image compression, inpainting and compressive sensing (Aharon et al., 2006; Thiagarajan et al., 2011). The general sparse representation and dictionary learning optimization can posed as shown in (Aharon et al., 2006), i.e.,

$$\min_{\mathbf{\Psi},\mathbf{A}} \|\mathbf{Y} - \mathbf{\Psi}\mathbf{A}\|_F^2 \text{ subject to } \|\mathbf{a}_i\|_0 \leqslant s, \quad \forall i \quad \text{and } \|\psi_j\|_2 = 1, \quad \forall j, \tag{2}$$

where s is the sparsity (the maximum number of non-zero elements) of the representation of a training vector, $\|.\|_F$ is the Frobenius norm, $\|\cdot\|_0$ is the ℓ_0 norm and $\|\cdot\|_2$ is the ℓ_2 norm. We will use uppercase bold symbols for denoting either a set of vectors or a matrix and this will be clear from the context.

In the *K*-means clustering problem, we assume that each coefficient vector is 1-sparse, i.e., has exactly one non-zero coefficient and the coefficient value is 1. In the *K*-hyperline clustering problem, we still assume a sparsity of 1 for the coefficient vector, but the value of the non-zero coefficient itself is unconstrained. This can be obtained by using the value of $s = 1$ in (2). The columns of $\mathbf{\Psi}$, given by the set $\{\psi_j\}_{j=1}^K$ are the K normalized cluster centroids. Each normalized cluster centroid denotes a 1-D linear subspace that passes through the origin. Let us denote the data in the jth cluster by the matrix $\mathbf{Y}_j = [\mathbf{y}_i]_{i\in\mathcal{C}_j}$, where the membership set $\mathcal{C}_j$ contains the training vector indices corresponding to the cluster j. The singular value decomposition (SVD) (Golub and Loan, 1996) of $\mathbf{Y}_j = \mathbf{U}_j\mathbf{\Upsilon}_j\mathbf{V}_j^T$, where $\mathbf{U}_j$ and $\mathbf{V}_j$ are orthonormal matrices of size $M \times M$ and $|\mathcal{C}_j| \times |\mathcal{C}_j|$ respectively. $\mathbf{\Upsilon}_j$ is a diagonal matrix with the singular values arranged in descending order. The columns of $\mathbf{U}_j$ and $\mathbf{V}_j$ are respectively the singular vectors for the columns and rows of the matrix $\mathbf{Y}_j$. The first column of $\mathbf{U}_j$ is the singular vector corresponding to the largest singular value of $\mathbf{Y}_j$, and it is the centroid of cluster j. This clustering algorithm is suited for sparse component analysis, where an observable data matrix is separated into a sparse linear combination of hidden sources (He et al., 2009). Sparse component analysis is particularly well suited for underdetermined blind source separation, where the number of observations is less than the number of sources. In our earlier work (Thiagarajan et al., 2011), we have used *K*-hyperline clustering

* Corresponding author. Tel.: +1 480 297 6883; fax: +1 480 965 8325.
E-mail address: knatesan@asu.edu (K.N. Ramamurthy).

0167-8655/$ - see front matter © 2011 Elsevier B.V. All rights reserved.
doi:10.1016/j.patrec.2011.03.005

for learning a multilevel dictionary that is useful in various applications involving sparse approximations.

For a wide class of distortion measures, known as Bregman divergences, optimal hard as well as soft clustering schemes have been analyzed in (Banerjee et al., 2005b). The classical K-means algorithm is a hard clustering approach when the Bregman divergence measure used is the squared Euclidean distance. Many clustering algorithms are equivalent to vector quantization procedures. For example, the K-means procedure is also referred to as the generalized Lloyd algorithm for vector quantizer design. The local optimality of K-means can be shown using the Lloyd's conditions for the optimality of a vector quantizer (Gersho and Gray, 1992). Clustering algorithms based on Bregman divergence measures are shape quantizers, where each training vector is assigned to a cluster centroid (shape) and the cluster centroids are obtained using conditional expectations of training vectors (Banerjee et al., 2005a). However, in K-hyperline clustering a coefficient (gain) unique to the training vector is computed in addition to assigning the training vector to a centroid. Therefore, K-hyperline clustering is similar, but not identical, to a shape-gain quantization scheme (Gersho and Gray, 1992) because the gain values are unquantized. In this work, we show that the K-hyperline clustering algorithm proposed in (He et al., 2009) converges to a locally optimal solution. The local optimality is proved by (a) showing that the algorithm satisfies the Lloyd's conditions for an optimal vector quantizer and, (b) by posing the problem of dictionary learning as an Expectation-Maximization (EM) procedure (Dempster et al., 1977) and showing that K-hyperline clustering is a constrained version of the EM procedure.

The general idea behind stability of a clustering algorithm is that the algorithm should produce cluster centroids that are not significantly different when different i.i.d. training sets from the same probability space are used for training (Ben-David et al., 2006; Ben-David et al., 2007; Rakhlin and Caponnetto, 2007). The authors in (Ben-David et al., 2006) show that a clustering algorithm is stable if there is a unique minimizer to the objective function. This notion is extended in (Ben-David et al., 2007) to characterize the stability of the K-means clustering algorithm, based on the number of optimal solutions to the underlying clustering problem. A different idea of stability is used in (Rakhlin and Caponnetto, 2007) to prove that the K-means clustering is stable. When there is no unique global minimizer to the objective function, it is shown in (Rakhlin and Caponnetto, 2007) that K-means is stable with respect to a change in $o(\sqrt{T})$ samples between two i.i.d. training sets of T samples each, as $T \to \infty$. This is performed by (a) posing the K-means clustering as an empirical risk minimization problem over a class of distortion functions, (b) identifying the class as uniform Donsker by finding the covering number with respect to the supremum norm for the class and, (c) proving the stability of the cluster centroids from the stability of the distortion function class. In this paper, we follow the line of reasoning given in (Rakhlin and Caponnetto, 2007), to prove the stability of K-hyperline clustering with respect to a change in $o(\sqrt{T})$ samples. We also show that the K-hyperline clustering becomes unstable when some training vectors have their Euclidean norm close to zero. Because of the different geometries, the proofs of stability are substantially different for K-means and K-hyperline clustering algorithms.

The rest of this paper is organized as follows. In Section 2 we describe the algorithm, derive the covering number for the distortion function class and show that the cluster centroids tessellate the space into convex Voronoi regions. The convergence and optimality of the clustering algorithm is proved in Section 3. Section 4 presents the proof of stability of the K-hyperline clustering and Section 5 concludes the paper.

2. The K-hyperline clustering algorithm

Let us assume that the data $\mathcal{Y}$ lies in $\mathbb{R}^M$ and define the probability space $(\mathcal{Y}, \mathbf{\Sigma}, P)$, where P is an unknown probability measure. The training samples, $\{\mathbf{y}_i\}_{i=1}^T$, are T i.i.d. realizations from the probability space. We also define an empirical probability measure P_T that assigns the mass T^{-1} to each of the T training samples (van de Geer, 2000). The goal of the clustering is to find K partitions of the training data that results in minimum total distortion.

The K-hyperline clustering algorithm is similar to the K-means algorithm and it is an alternating minimization problem that proceeds in two stages after initialization: the cluster assignment and the cluster centroid update stages. In the cluster assignment stage, the training vector $\mathbf{y}_i$ is assigned to a cluster j based on the minimum distortion criteria, $\mathcal{H}(\mathbf{y}_i) = \text{argmin}_j d(\mathbf{y}_i, \boldsymbol{\psi}_j)$, which is equivalent to $\mathcal{H}(\mathbf{y}_i) = \text{argmax}_j |\mathbf{y}_i^T \boldsymbol{\psi}_j|$. Here, $\mathcal{H}(\cdot)$ is the membership function that returns the cluster index of a training vector and the distortion measure is

$$d(\mathbf{y}, \boldsymbol{\psi}) = \|\mathbf{y} - \boldsymbol{\psi}(\mathbf{y}^T \boldsymbol{\psi})\|_2^2, \tag{3}$$

where $\boldsymbol{\psi}$ is assumed to be normalized. Since the centroids in K-hyperline clustering are 1-D linear subspaces, the distortion measure indicates the squared length of the residual obtained after orthogonal projection of the data $\mathbf{y}$ onto its centroid $\boldsymbol{\psi}$. The membership set $\mathcal{C}_j = \{i | \mathcal{H}(\mathbf{y}_i) = j\}$ contains training vector indices corresponding to the cluster j. Ties in the cluster assignment stage are broken arbitrarily. Based on the cluster assignment, the updated cluster centroids can be obtained as described in Section 2.1.

2.1. Cluster centroid

Given the set $\mathcal{C}_j$, the j^{th} cluster centroid is updated as $\boldsymbol{\psi}_j = \text{argmin}_{\boldsymbol{\psi}} \mathbf{E}_{P_T} [d(\mathbf{y}, \boldsymbol{\psi}) | \mathcal{H}(\mathbf{y}) = j]$. This can also be expressed using the following equation:

$$\boldsymbol{\psi}_j = \text{argmin}_{\boldsymbol{\psi}} \sum_{i \in \mathcal{C}_j} \|\mathbf{y}_i - \boldsymbol{\psi}(\mathbf{y}_i^T \boldsymbol{\psi})\|_2^2. \tag{4}$$

Consider the matrix $\mathbf{Y}_j = [\mathbf{y}_i]_{i \in \mathcal{C}_j}$ and the SVD of $\mathbf{Y}_j = \mathbf{U}_j \boldsymbol{\Upsilon}_j \mathbf{V}_j^T$, where $\boldsymbol{\Upsilon}_j$ is a diagonal matrix with singular values arranged in descending order. The solution to (4) is obtained by taking $\boldsymbol{\psi}_j$ as the first column of $\mathbf{U}_j$. Note that the K-hyperline clustering is not a Bregman divergence based clustering scheme since the centroid is not the conditional expectation of the training vectors (Banerjee et al., 2005a).

Let us assume a generative model for the training vectors $\{\mathbf{y}_i\}_{i \in \mathcal{C}_j}$ in cluster j as, $\mathbf{y}_i = a_{ji}\, \boldsymbol{\psi}_j + \mathbf{n}_i$, where $\mathbf{n}_i$ are i.i.d. realizations from $\mathcal{N}(0, \sigma^2 \mathbf{I})$ and a_{ji} are arbitrary coefficients. For this model, the Maximum Likelihood (ML) estimate of the cluster centroid $\boldsymbol{\psi}_j$ is obtained using SVD as described earlier and $a_{ji} = \mathbf{y}_i^T \boldsymbol{\psi}_j$. If we constrain $a_{ji} = 1$, the ML estimate of $\boldsymbol{\psi}_j$ is the mean of $\{\mathbf{y}_i\}_{i \in \mathcal{C}_j}$, which is similar to the case of K-means clustering. Because of the flexibility of incorporating coefficients, the K-hyperline clustering typically achieves a lesser residual error than the K-means clustering. This motivates the use of SVD based learning algorithms such as the ones proposed in (Aharon et al., 2006; Thiagarajan et al., 2008; Thiagarajan et al., 2011), for sparse representations.

2.2. Distortion function for clustering

Mathematically, the K-hyperline clustering is a problem of finding normalized centroids that minimize the total distortion,

$$D(\mathcal{H}) = \frac{1}{T} \sum_{j=1}^{K} \sum_{i \in \mathcal{C}_j} d(\mathbf{y}_i, \boldsymbol{\psi}_j) = \frac{1}{T} \sum_{j=1}^{K} \sum_{i \in \mathcal{C}_j} \mathbf{y}_i^T \left(\mathbf{I} - \boldsymbol{\psi}_j \boldsymbol{\psi}_j^T \right) \mathbf{y}_i. \tag{5}$$