Improving Parametric PCA Using KL-divergence Between Gaussian-Markov Random Field Models

Alexandre L. M. Levada(✉)

Computing Department, Federal University of São Carlos, São Carlos, SP, Brazil
alexandre.levada@ufscar.br

Abstract. Parametric PCA is a dimensionality reduction based metric learning method that uses the Bhattacharyya and Hellinger distances between Gaussian distributions estimated from local patches of the KNN graph to build the parametric covariance matrix. Later on, PCA-KL, an entropic PCA version using the symmetrized KL-divergence (relative entropy) was proposed. In this paper, we extend this method by replacing the Gaussian distribution by a Gaussian-Markov random field model. The main advantage is the incorporation of the spatial dependence by means of the inverse temperature parameter. A closed form expression for the KL-divergence is derived, allowing fast computation and avoiding numerical simulations. Results with several real datasets show that the proposed method is capable of improving the average classification performance in comparison to PCA-KL and some state-of-the-art manifold learning algorithms, such as t-SNE and UMAP.

Keywords: Dimensionality reduction · Metric learning · PCA · KL-divergence · Markov random fields

1 Introduction

Metric learning is concerned with the process of finding suitable and adaptive distance functions prior to pattern classification problems. [9]. A powerful strategy for achieving this goal consists in finding a more compact and meaningful representation for the observed data, which is the mathematical framework of dimensionality reduction methods [19,20].

Linear dimensionality reduction methods have been developed throughout many areas of science such as statistics, optimization, machine learning, and other applied fields for over a century, and these methods have become powerful mathematical tools for analyzing high dimensional and noisy data. Part of the success of the linear methods comes from the fact that they have simple geometric interpretations and typically attractive computational properties [3].

Nonlinear dimensionality reduction involves finding low-dimensional representations in high-dimensional space. This problem arises naturally when analyzing data like hyperspectral imagens, human faces, speech waveforms and handwritten characters. Linear algorithms often fail to capture the hidden non-linear

© Springer Nature Switzerland AG 2021
O. Gervasi et al. (Eds.): ICCSA 2021, LNCS 12950, pp. 55–69, 2021.
https://doi.org/10.1007/978-3-030-86960-1_5

representation of the data in the presence of non-Euclidean geometry. It has been found that manifolds are used to explain how our visual perception and learning work by recognizing the variability of perceptual stimuli and other types of high-dimensional data [10,12,15].

Manifold learning algorithms explore intrinsic geometric properties of the datasets to build a low dimensional representation using non-Euclidean distances [16]. However, one limitation with these methods is that it is not clear how to deal with out-of-sample problems, that is, how to find the low dimensional representation of samples that do not belong to the training set.

Recently, Parametric PCA has been proposed as an alternative to dimensionality reduction based metric learning algorithms. The basic idea consists in using the Bhattacharyya and Hellinger distances between univariate Gaussian distributions learned from the local patches along the KNN graph to compute a surrogate for the covariance matrix [6]. Later on, PCA-KL has been proposed as a generalization of Parametric PCA, in which the symmetrized KL-divergence (relative entropy) is employed to build the entropic covariance matrix [8].

In this paper, we propose a contextual version of PCA-KL that has two main contributions in comparison with regular PCA-KL: 1) to capture the dependencies between pieces of data inside a local patch, instead of using a Gaussian distribution, we use a Gaussian-Markov random field to model the intra-patch correlations; 2) to avoid k-regular graphs, instead of using the KNN rule, we use the ϵ-neighborhood rule, allowing patches of variable sizes along the graph. Another positive aspect of the proposed method is that by replacing a point-wise similarity measure (Euclidean distance) by a path-based one (symmetric KL-divergence), we can make the whole process more robust to the presence of noise and outliers in the observations.

The remaining of the paper is organized as follows: Sect. 2 and its subsections describe the proposed method in details, from the random field model and the computation of the KL-divergence to the contextual PCA-KL algorithm. Section 3 shows the experiments and the obtained results from an objective comparison of the proposed method with several dimensionality reduction methods in different real world datasets. Finally, Sect. 4 brings our conclusions, final remarks and future directions for research.

2 The Proposed Method

In this section, we describe the proposed method in details, from the definition of the statistical model to the derivation of a closed-form expression for the KL-divergence between pairwise GMRF's and the contextual PCA-KL algorithm.

2.1 The Random Field Model

Pairwise Gaussian-Markov random fields (GMRF's) are mathematical structures particularly suitable to study dependent continuous random variables by means of non-linear interactions between neighboring variables that are part of a grid.

The main advantage of this model in comparison to other random field models is related to mathematical tractability. Often, we cannot derive closed-form expressions for statistical divergences, such as the Kullback-Leibler divergence, due to the presence of expected values. In pairwise GMRF's, it is possible to compute these measures analytically, which reduces the computational burden. Moreover, by invoking the Hammersley-Clifford theorem [4], which states the equivalence between Gibbs random fields (global models) and Markov random fields (local models), we can characterize an isotropic pairwise Gaussian-Markov random field by the set of local conditional density functions, given by:

$$p\left(x_i|\eta_i,\vec{\theta}\right) = \frac{1}{\sqrt{2\pi\sigma^2}} exp\left\{-\frac{1}{2\sigma^2}\left[x_i - \mu - \beta\sum_{j\in\eta_i}(x_j-\mu)\right]^2\right\} \tag{1}$$

where η_i denotes the neighborhood system comprised by the nearest neighbors of x_i, $\vec{\theta} = (\mu, \sigma^2, \beta)$ denotes the vector of model parameters, with μ and σ^2 being the expected value (mean) and the variance of the random variables in the grid, and β representing the inverse temperature, which encodes the correlation between the variables in η_i. Note that when $\beta = 0$, the model degenerates a the simple Gaussian distribution. The main advantage of using the local model is that we avoid the joint Gibbs distribution and the need of intense Markov Chain Monte Carlo (MCMC) simulations. In this work, we use a pairwise GMRF to model the dependency between Gaussian random variables in a 1-D grid. Figure 1 illustrates different neighborhood systems, from first to third order. Note that the number of neighbors is twice the order of the neighborhood system.

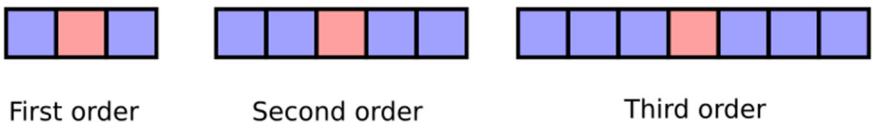

Fig. 1. Several neighborhood systems defined in a 1-D grid: from left to right, we have first, second and third order systems.

2.2 The KL-divergence Between GMRF Models

The entropy of a random variable x is the expected value of the self-information:

$$H(p) = -\int p(x)[log\ p(x)]dx = -E\left[log\ p(x)\right] \tag{2}$$

where $p(x)$ is the probability density function (pdf) of x. For a Gaussian random variable x, distributed as $N(\mu, \sigma^2)$, the entropy can be computed as:

$$H(p) = \frac{1}{2}log\ (2\pi\sigma^2) + \frac{1}{2\sigma^2}E[(x-\mu)^2] = \frac{1}{2}\left(1 + log\ (2\pi\sigma^2)\right) \tag{3}$$

The cross-entropy between two random variables with pdf's $p(x)$ and $q(x)$ is defined by:

$$H(p,q) = -\int p(x)[log\ q(x)]dx \tag{4}$$

The Kullback-Leibler divergence is the relative entropy, that is, the difference between the cross-entropy of $p(x)$ and $q(x)$ and the entropy of $p(x)$:

$$\begin{aligned} D_{KL}(p,q) = H(p,q) - H(p) &= -\int p(x)[log\ q(x)]dx + \int p(x)[log\ p(x)]dx \\ &= \int p(x) log\left(\frac{p(x)}{q(x)}\right) dx = E_p\left[log\left(\frac{p(x)}{q(x)}\right)\right] \end{aligned} \tag{5}$$

It should be mentioned that the relative entropy is always non-negative, that is, $D_{KL}(p,q) \geq 0$, being equal to zero if, and only if, $p(x) = q(x)$. Let $p(x)$ and $q(x)$ be univariate Gaussian densities, $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$. Then, the KL-divergence between these two pdf's is:

$$\begin{aligned} D_{KL}(p,q) &= E_p\left[-log\ \sigma_1 - \frac{1}{2\sigma_1^2}(x-\mu_1)^2 + log\ \sigma_2 - \frac{1}{2\sigma_2^2}(x-\mu_2)^2\right] \\ &= log\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2\sigma_2^2}E_p[(x-\mu_2)^2] - \frac{1}{2\sigma_1^2}E_p[(x-\mu_1)^2] \end{aligned} \tag{6}$$

From basic statistics, it is known that:

$$E_p[(x-\mu_1)^2] = \sigma_1^2 \tag{7}$$

$$E_p[(x-\mu_2)^2] = E[x^2] - 2E[x]\mu_2 + \mu_2^2 \tag{8}$$

$$E[x^2] = Var[x] + E^2[x] = \sigma_1^2 + \mu_1^2 \tag{9}$$

which leads to:

$$\begin{aligned} D_{KL}(p,q) &= log\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2\sigma_2^2}(\sigma_1^2 + \mu_1^2 - 2\mu_1\mu_2 + \mu_2^2) - \frac{1}{2} \\ &= log\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{\sigma_1^2 + (\mu_1-\mu_2)^2}{2\sigma_2^2} - \frac{1}{2} \end{aligned} \tag{10}$$

Note that $D_{KL}(p,q) \neq D_{KL}(q,p)$, that is, the relative entropy is not symmetric. The symmetrized KL-divergence between $p(x)$ and $q(x)$ is:

$$\begin{aligned} D_{KL}^{sym}(p,q) &= \frac{1}{2}[D_{KL}(p,q) + D_{KL}(q,p)] \\ &= \frac{1}{4}\left[\frac{\sigma_1^2 + (\mu_1-\mu_2)^2}{\sigma_2^2} + \frac{\sigma_2^2 + (\mu_1-\mu_2)^2}{\sigma_1^2} - 2\right] \\ &= \frac{1}{4\sigma_1^2\sigma_2^2}\left[\left(\sigma_1^2 - \sigma_2^2\right)^2 + (\mu_1-\mu_2)^2\left(\sigma_1^2 + \sigma_2^2\right)\right] \end{aligned} \tag{11}$$

Similarly, the KL-divergence between two GMRF's with parameter vectors $\vec{\theta}_1 = (\mu_1, \sigma_1^2, \beta_1)$ and $\vec{\theta}_2 = (\mu_2, \sigma_2^2, \beta_2)$ is given by:

$$D_{KL}(p,q) = log\left(\frac{\sigma_2}{\sigma_1}\right) + \frac{1}{2\sigma_2^2} E_p\left\{\left[x - \mu_2 - \beta_2 \sum_{j\in\eta_i}(x_j - \mu_2)\right]^2\right\} - \frac{1}{2\sigma_1^2} E_p\left\{\left[x - \mu_1 - \beta_1 \sum_{j\in\eta_i}(x_j - \mu_1)\right]^2\right\} \tag{12}$$

It is possible to expand the second expected value in Eq. (12) as:

$$\begin{aligned} &E_p\left\{\left[x - \mu_1 - \beta_1 \sum_{j\in\eta_i}(x_j - \mu_1)\right]^2\right\} \\ &= E_p\left\{(x-\mu_1)^2 - 2\beta_1 \sum_{j\in\eta_i}(x_i - \mu_1)(x_j - \mu_1) + \beta_1^2 \sum_{j\in\eta_i}\sum_{k\in\eta_i}(x_j - \mu_1)(x_k - \mu_1)\right\} \\ &= \sigma_1^2 - 2\beta_1 \sum_{j\in\eta_i}\sigma_{ij} + \beta_1^2 \sum_{j\in\eta_i}\sum_{k\in\eta_i}\sigma_{jk} \end{aligned} \tag{13}$$

where σ_{ij} denotes the covariance between the central variable x_i and a neighboring variable x_j and σ_{jk} denotes the covariance between two different variables belonging to the same neighborhood system η_i. Now, we expand the first expected value as:

$$\begin{aligned} &E_p\left\{\left[x - \mu_2 - \beta_2 \sum_{j\in\eta_i}(x_j - \mu_2)\right]^2\right\} \\ &= E_p\left\{(x-\mu_2)^2 - 2\beta_2 \sum_{j\in\eta_i}(x_i - \mu_2)(x_j - \mu_2) + \beta_2^2 \sum_{j\in\eta_i}\sum_{k\in\eta_i}(x_j - \mu_2)(x_k - \mu_2)\right\} \end{aligned} \tag{14}$$

We need to break Eq. (14) in three distinct expected values in order to simplify it. The first one remains identical to the corresponding term in the Gaussian case:

$$E_p\left\{(x-\mu_2)^2\right\} = \sigma_1^2 + (\mu_1 - \mu_2)^2 \tag{15}$$

The second expected value is given by:

$$\begin{aligned} -2\beta_2 \sum_{j\in\eta_i} E_p\left\{(x_i - \mu_2)(x_j - \mu_2)\right\} &= -2\beta_2 \sum_{j\in\eta_i} E_p\left\{x_i x_j - \mu_2 x_i - \mu_2 x_j + \mu_2^2\right\} \\ &= -2\beta_2 \sum_{j\in\eta_i}\left[\sigma_{ij}^{(p)} + (\mu_1 - \mu_2)^2\right] \end{aligned} \tag{16}$$

where $\sigma_{ij}^{(p)}$ denotes the covariance between x_i and x_j from the covariance matrix Σ_p. In a similar way, the third expected value leads to:

$$\beta_2^2 \sum_{j\in\eta_i}\sum_{k\in\eta_i} E_p\left\{(x_j-\mu_2)(x_k-\mu_2)\right\} = \beta_2^2 \sum_{j\in\eta_i}\sum_{k\in\eta_i}\left[\sigma_{jk}^{(p)} + (\mu_1-\mu_2)^2\right] \quad (17)$$

Thus, the KL-divergence can be expressed as:

$$\begin{aligned} D_{KL}(p,q) = log\left(\frac{\sigma_2}{\sigma_1}\right) - \frac{1}{2\sigma_1^2}\left[\sigma_1^2 - 2\beta_1\sum_{j\in\eta_i}\sigma_{ij} + \beta_1^2\sum_{j\in\eta_i}\sum_{k\in\eta_i}\sigma_{jk}\right] \quad (18)\\ + \frac{1}{2\sigma_2^2}\left\{\sigma_1^2 + (\mu_1-\mu_2)^2 - 2\beta_2\sum_{j\in\eta_i}\left[\sigma_{ij}^{(p)} + (\mu_1-\mu_2)^2\right]\right.\\ \left. +\beta_2^2\sum_{j\in\eta_i}\sum_{k\in\eta_i}\left[\sigma_{jk}^{(p)} + (\mu_1-\mu_2)^2\right]\right\} \end{aligned}$$

By simplifying the last term of the previous equation, we have the final expression for the KL-divergence as:

$$\begin{aligned} D_{KL}(p,q) = log\left(\frac{\sigma_2}{\sigma_1}\right) - \frac{1}{2\sigma_1^2}\left[\sigma_1^2 - 2\beta_1\sum_{j\in\eta_i}\sigma_{ij}^{(p)} + \beta_1^2\sum_{j\in\eta_i}\sum_{k\in\eta_i}\sigma_{jk}^{(p)}\right] \quad (19)\\ + \frac{1}{2\sigma_2^2}\left\{\left[\sigma_1^2 - 2\beta_2\sum_{j\in\eta_i}\sigma_{ij}^{(p)} + \beta_2^2\sum_{j\in\eta_i}\sum_{k\in\eta_i}\sigma_{jk}^{(p)}\right] + (\mu_1-\mu_2)^2(1-\Delta\beta_2)^2\right\} \end{aligned}$$

where Δ denotes the number of neighbors in the neighborhood system η_i. Note that, when the inverse temperatures are zero, that is, $\beta_1 = \beta_2 = 0$, $D_{KL}(p,q)$ becomes Eq. (10), which is the KL-divergence between two Gaussian pdf's.

By analogy, the expression for $D_{KL}(q,p)$ can be written as:

$$\begin{aligned} D_{KL}(q,p) = log\left(\frac{\sigma_1}{\sigma_2}\right) - \frac{1}{2\sigma_2^2}\left[\sigma_2^2 - 2\beta_2\sum_{j\in\eta_i}\sigma_{ij}^{(q)} + \beta_2^2\sum_{j\in\eta_i}\sum_{k\in\eta_i}\sigma_{jk}^{(q)}\right] \quad (20)\\ + \frac{1}{2\sigma_1^2}\left\{\left[\sigma_2^2 - 2\beta_1\sum_{j\in\eta_i}\sigma_{ij}^{(q)} + \beta_1^2\sum_{j\in\eta_i}\sum_{k\in\eta_i}\sigma_{jk}^{(q)}\right] + (\mu_1-\mu_2)^2(1-\Delta\beta_1)^2\right\} \end{aligned}$$

Finally, after some algebraic manipulations, the symmetrized KL-divergence between two pairwise Gaussian-Markov random field models is given by:

$$
\begin{aligned}
D_{KL}^{sym}(p,q) &= \frac{1}{2}[D_{KL}(p,q) + D_{KL}(q,p)] \\
&= \frac{1}{4\sigma_1^2\sigma_2^2}\left\{\left(\sigma_1^2-\sigma_2^2\right)^2 - 2(\beta_2\sigma_1^2-\beta_1\sigma_2^2)\sum_{j\in\eta_i}\left(\sigma_{ij}^{(p)}-\sigma_{ij}^{(q)}\right)\right. \\
&+(\beta_2^2\sigma_1^2-\beta_1^2\sigma_2^2)\sum_{j\in\eta_i}\sum_{k\in\eta_i}\left(\sigma_{jk}^{(p)}-\sigma_{jk}^{(q)}\right) \qquad (21) \\
&\left.+\left(\mu_1-\mu_2\right)^2\left[\sigma_1^2(1-\Delta\beta_2)^2+\sigma_2^2(1-\Delta\beta_1)^2\right]\right\}
\end{aligned}
$$

Once again, note that when $\beta_1 = \beta_2 = 0$, the expression becomes the symmetrized KL-divergence between two Gaussian random variables, as indicated by Eq. (11).

2.3 Maximum Pseudo-likelihood Estimation

The KL-divergence between two GMRF's is a function of the model parameters: mean, variances, covariances and the inverse temperature. In our experiments, the GMRF's parameters μ and Σ, the covariance matrix of the patches, are both estimated via maximum likelihood. However, maximum likelihood estimation of the inverse temperature parameter is intractable, because we need to compute the partition function in the joint Gibbs distribution. To overcome this limitation, we perform maximum pseudo-likelihood estimation, which is based on the conditional independence principle [2]. In the pairwise Gaussian-Markov random field model, we have to maximize the logarithm of the product of all the local conditional density functions:

$$
log\ L\left(\vec{\theta};\mathbf{X}\right) = -\frac{n}{2}log\left(2\pi\sigma^2\right) - \frac{1}{2\sigma^2}\sum_{i=1}^{n}\left[x_i-\mu-\beta\sum_{j\in\eta_i}\left(x_j-\mu\right)\right]^2 \qquad (22)
$$

If we differentiate Eq. (22) with respect to β and properly solve the pseudo-likelihood equation, we have [7]:

$$
\hat{\beta}_{MPL} = \frac{\sum_{i=1}^{n}\left[\sum_{j\in\eta_i}\left(x_i-\mu\right)\left(x_j-\mu\right)\right]}{\sum_{i=1}^{n}\left[\sum_{j\in\eta_i}\left(x_j-\mu\right)\right]^2} \qquad (23)
$$

Another simpifying assumption is that in our case, the random field is defined on a 1-D grid, where the size of the neighborhood system is fixed ($\Delta = 2, 4, 6, ...$). Hence, the maximum pseudo-likelihood estimator for the inverse temperature can be expressed in terms of the covariances:

$$\hat{\beta}_{MPL} = \frac{\sum_{j \in \eta_i} \sigma_{ij}}{\sum_{j \in \eta_i} \sum_{k \in \eta_i} \sigma_{jk}} \tag{24}$$

which means that we can use the covariance matrix of the patches extracted from the random sample.

2.4 Numerical Computations

In order to speed up the numerical computations, we propose to express the equations for the maximum pseudo-likelihood estimator of the inverse temperature parameter and the KL-divergence between two GMRF's using tensor notation. First, note that given a random sample of a Gaussian variable with size n, denoted by $X = \{x_1, x_2, ..., x_n\}$, we can build a $n \times (\Delta + 1)$ dataset in which the i-th row represents the i-th patch. For instance, considering a first order neighborhood system, that is, $\Delta = 2$, we have $p_i = [x_{i-1}, x_i, x_{i+1}]$. Then, we compute the covariance matrix of these patches, denoted by Σ. From this covariance matrix, we extract two main components: 1) a vector $\vec{\rho}$, composed by the elements of the central row of Σ, excluding the middle one, which denotes the variance of x_i (we want only the covariances between x_i and x_j, for $j \neq i$; and 2) a sub-matrix Σ^-, obtained by removing the central row and central column from Σ (we want only the covariances between $x_j \in \eta_i$ and $x_k \in \eta_i$). Figure 2 shows the decomposition of the covariance matrix Σ into the sub-matrix Σ^- and the vector $\vec{\rho}$ in a fourth order neighborhood system ($\Delta = 8$).

We can now use Kronecker products to rewrite the Fisher information matrices in a tensor notation, providing a computationally efficient way to compute the maximum pseudo-likelihood estimator of the inverse temperature parameter as:

$$\hat{\beta}_{MPL} = \frac{\|\vec{\rho}\|_+}{\|\Sigma^-\|_+} \tag{25}$$

where $\|A\|_+$ denotes the summation of the elements of the vector/matrix A. Using the same logic, the expression for the symmetrized KL-divergence between two GMRF's can be expressed as:

$$\begin{aligned} D_{KL}^{sym}(p,q) = \frac{1}{4\sigma_1^2\sigma_2^2} \Bigg\{ & \left(\sigma_1^2 - \sigma_2^2\right)^2 - 2(\beta_2\sigma_1^2 - \beta_1\sigma_2^2) \left(\left\| \vec{\rho}^{(p)} \right\|_+ - \left\| \vec{\rho}^{(q)} \right\|_+ \right) \\ & + (\beta_2^2\sigma_1^2 - \beta_1^2\sigma_2^2) \left(\left\| \Sigma_{(p)}^- \right\|_+ - \left\| \Sigma_{(q)}^- \right\|_+ \right) \\ & + (\mu_1 - \mu_2)^2 \left[\sigma_1^2(1 - \Delta\beta_2)^2 + \sigma_2^2(1 - \Delta\beta_1)^2 \right] \Bigg\} \end{aligned} \tag{26}$$

where $\vec{\rho}^{(p)}$ and $\Sigma_{(p)}^-$ are extracted from the covariance matrix of the first GMRF and $\vec{\rho}^{(q)}$ and $\Sigma_{(q)}^-$ are extracted from the covariance matrix of the second GMRF.

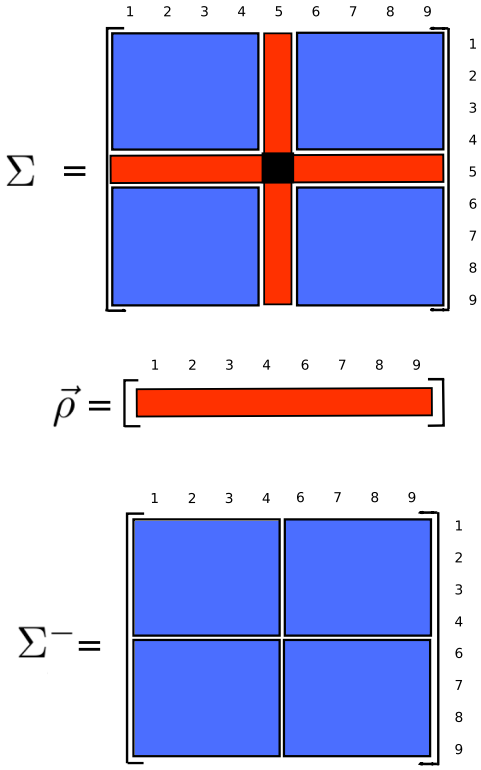

Fig. 2. Decomposition of Σ **into** Σ^{-} **and** $\vec{\rho}$ **on a fourth order neighborhood system** ($\Delta = 8$)**.** By rewriting maximum pseudo-likelihood estimator of the inverse temperature parameter and the KL-divergence between two GMRF's in terms of Kronocker products, we speed up the numerical computations.

2.5 PCA-KL with Pairwise GMRF's

The contextual PCA-KL using pairwise GMRF's proposed here is inspired by Parametric PCA, a recent dimensionality reduction based metric learning algorithm that computes the entropic covariance matrix of the data through information-theoretic divergences estimated in local patches along the KNN graph [6]. Previous works proposed extensions to Parametric PCA by replacing the Bhatthacharyya and Hellinger distances by the symmetrized KL-divergence [8]. The main difference between the proposed method and PCA-KL concerns the statistical model employed to fit the data.

Let $X = \{\vec{x}_1, \vec{x}_2, ..., \vec{x}_n\}$, where $\vec{x}_i \in R^m$, represent our input data matrix and $G = (V, E)$ be the ϵ-neighborhood graph induced from X by linking each pair of samples $\vec{x}_i$ and $\vec{x}_j$ if $d_E(\vec{x}_i, \vec{x}_j) < \epsilon$, where $d_E(.,.)$ is the Euclidean distance. Hence, we can define a patch P_i as the set composed by a sample $\vec{x}_i$ and its

neighbors in the graph. Let k_i denote the number of neighbors of $\vec{x}_i$ in G. Then, a patch P_i is a $(k+1) \times m$ matrix:

$$P_i = \begin{bmatrix} \vec{x}_i \\ \vec{x}_{i1} \\ \vec{x}_{i2} \\ \vec{x}_{i3} \\ \vdots \\ \vec{x}_{ik_i} \end{bmatrix} = \begin{bmatrix} \vec{x}_i(1) & \vec{x}_i(2) & \cdots & \vec{x}_i(m) \\ \vec{x}_{i1}(1) & \vec{x}_{i1}(2) & \cdots & \vec{x}_{i1}(m) \\ \vec{x}_{i2}(1) & \vec{x}_{i2}(2) & \cdots & \vec{x}_{i2}(m) \\ \vec{x}_{i3}(1) & \vec{x}_{i3}(2) & \cdots & \vec{x}_{i3}(m) \\ \vdots & \vdots & \ddots & \vdots \\ \vec{x}_{ik_i}(1) & \vec{x}_{ik_i}(2) & \cdots & \vec{x}_{ik_i}(m) \end{bmatrix}$$

where $\vec{x}_{i1}, \vec{x}_{i2}, \ldots, \vec{x}_{ik_i}$ denote the k_i neighbors of $\vec{x}_i$ in the ϵ-neighborhood graph. The idea is to consider that each column of the matrix P_i is a sample of a pairwise Gaussian-Markov random field, characterized a set of local conditional density functions. In the proposed method, we estimate the parameter vetors $\vec{\theta}_k = (\mu_k, \Sigma_k, \beta_k)$ for $k = 1, 2, ..., \ldots, m$ for each patch of the ϵ-neighborhood graph.

The proposed contextual PCA-KL method maps each patch P_i to a m-dimensional vector composed by 3-dimensional tuples, where each tuple j contains the estimatives of the model parameters. As we have m distinct features, we will have exactly m tuples. The parametric feature vector $\vec{p}_i$ for the patch P_i is defined as:

$$\vec{p}_i = \left[\vec{\theta}_1^{(i)}, \vec{\theta}_2^{(i)}, ..., \vec{\theta}_m^{(i)}\right] \tag{27}$$

where each component is a tuple of parameters:

$$\vec{\theta}_j^{(i)} = \left(\mu_j^{(i)}, \Sigma_j^{(i)}, \beta_j^{(i)}\right) \tag{28}$$

The space spanned by $\vec{p}_i$, for $i = 1, 2, ..., n$, is the parametric feature space. The sample average of all $\vec{p}_i$'s represents the mean distribution and defines the centroid of this space:

$$\tilde{\vec{p}} = \frac{1}{n} \sum_{i=1}^{n} \vec{p}_i \tag{29}$$

In PCA-KL, the difference between two parametric vectors is computed by the symmetrized KL-divergence between each one of the tuples, that is:

$$\vec{p}_i - \vec{p}_j = \left[d_{KL}(\vec{\theta}_1^{(i)}, \vec{\theta}_1^{(j)}), ..., d_{KL}(\vec{\theta}_m^{(i)}, \vec{\theta}_m^{(j)})\right] = \vec{d}_{KL}\left(\vec{p}_i, \vec{p}_j\right) \tag{30}$$

For pairwise Gaussian-Markov random field models, a closed-form expression for the symmetrized KL-divergence is given by Eq. (21). Then, the entropic covariance matrix C is defined by:

$$C = E\left[\vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}})\ \vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}})^T\right] = \frac{1}{n-1} \sum_{i=1}^{n} \vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}})\ \vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}})^T \tag{31}$$

Note that with the proposed method it is really straightforward to deal with the out-of-sample problem, since instances that do not belong to the training

set, can be mapped to their new representation with the projection matrix. This is an advantage of PCA-KL in comparison to manifold learning algorithms, such as Laplacian Eigenmaps [1], t-SNE [18] and UMAP [11].

In the following, we present the an algorithm for unsupervised metric learning with contextual PCA-KL. The input is a data matrix $X_{m\times n}$, in which each column $\vec{x}_j \in R^m$ is a sample.

1. From the input data $\vec{x}_1, \vec{x}_2, ..., \vec{x}_n \in R^m$ build an undirected proximity graph using the ϵ-neighborhood rule;
2. For each patch, that is, $\vec{x}_i \cup \eta_i$, where η_i denotes the local neighborhood around $\vec{x}_i$, compute the sample mean of each feature. Additionally, estimate the covariance matrix of the patches and the inverse temperature for each feature ($k = 1, 2, ..., m$). At the end, this step generates for each patch, the following parametric vector:

$$\vec{p}_i = [(\mu_1, \Sigma_1, \beta_1), (\mu_2, \Sigma_2, \beta_2), ...(\mu_m, \Sigma_m, \beta_m)] \tag{32}$$

3. Compute the mean parametric vector for all patches, $\tilde{\vec{p}}$, which represents the average distribution, given all the dataset:

$$\tilde{\vec{p}} = \left[(\tilde{\mu}_1, \tilde{\Sigma}_1, \tilde{\beta}_1), (\tilde{\mu}_2, \tilde{\Sigma}_2, \tilde{\beta}_2), ..., (\tilde{\mu}_m, \tilde{\Sigma}_m, \tilde{\beta}_m)\right] \tag{33}$$

4. Compute the entropic covariance matrix C, a surrogate for the usual covariance matrix based in the symmetrized KL-divergence between each parametric vector $\vec{p}_i$ and the average distribution $\tilde{\vec{p}}$ as:

$$C = \frac{1}{n-1} \sum_{i=1}^{n} \vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}})\ \vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}})^T \tag{34}$$

where $\vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}})$ is a column vector of KL-divergences:

$$\vec{d}_{KL}(\vec{p}_i, \tilde{\vec{p}}) = \left[D_{KL}^{sym}\left(\vec{\theta}_1, \tilde{\vec{\theta}}_1\right), ..., D_{KL}^{sym}\left(\vec{\theta}_m, \tilde{\vec{\theta}}_m\right)\right]^T \tag{35}$$

with $\vec{\theta}_i = (\mu_i, \Sigma_i, \beta_i)$ and $\tilde{\vec{\theta}}_i = (\tilde{\mu}_i, \tilde{\Sigma}_i, \tilde{\beta}_i)$.
5. Select the $d < m$ eigenvectors associated to the d largest eigenvectors of the matrix C to compose the projection matrix W.
6. Project the data into the subspace spanned by W.

3 Experiments and Results

To evaluate the performance of the proposed contextual PCA-KL for dimensionality reduction based metric learning, we conducted a set of computational experiments to compare the average classification accuracies of eight different supervised classifiers (KNN, Naive Bayes, SVM, Decision Tree, Bayesian classifier under Gaussian hypothesis, Multilayer Perceptron, Gaussian Process Classifier and Random Forest) after dimensionality reduction to 2-D spaces. Our

hypothesis is that if the underlying distance functions are correctly learned during the feature extraction stage, we should observe a significant increase in the average scores when considering several real world datasets. We compared our method against PCA [5], Kernel PCA [14], PCA-KL (under Gaussian hypothesis) [8], ISOMAP [17], LLE [13], Laplacian Eigenmaps [1], t-SNE [18] and UMAP [11]. In all experiments, we selected 50% of the samples for training and 50% of the samples for testing. In the proposed method, we use a first order neighborhood system. Table 1 shows the average classification accuracies obtained from eight classifiers for several datasets after dimensionality reduction. The best result in a line is boldfaced. All 32 datasets, as well as detailed information regarding the number of instances, features and classes for each one of them, are freely available at openML.org.

To test if the results obtained by the proposed method are statistically superior, we performed a non-parametric Friedman test. There are strong evidences against the null hypothesis that all methods are identical ($p = 1.11 \times 10^{-16}$) for a significance level $\alpha = 0.01$. To check which groups are significantly different, we performed a Nemenyi post-hoc test. According to the test, for the same significance level, $\alpha = 0.01$, there are strong evidences that PCA-KL with GMRF's produced significantly higher accuracies than PCA ($p = 2.09 \times 10^{-11}$), Kernel PCA ($p = 3.11 \times 10^{-14}$), ISOMAP ($p = 1.80 \times 10^{-11}$), LLE ($p < 10^{-17}$), Laplacian Eigenmaps ($p < 10^{-17}$), t-SNE ($p = 3.28 \times 10^{-7}$) and UMAP ($p = 9.95 \times 10^{-10}$). Furthermore, there are evidences that PCA-KL with GMRF's produced significantly higher accuracies than traditional PCAKL ($p = 0.002$), which suggests that the Gaussian-Markov random field model is a better option in terms of dimensionality reduction based metric learning in these datasets. With the incorporation of the inverse temperature parameter in the model, we can capture relevant contextual information for data discrimination, something that is ignored by the usual Gaussian model, which assumes independence between the observations within a patch from the ϵ-neighborhood graph.

One possible limitation with the proposed method concerns the definition of the parameter ϵ (radius) that controls the patch size (number of neighbors of a given sample in the ϵ-neighborhood graph). Our experiments revealed that the classification accuracies are quite sensitive to changes in this parameter. In this study, we employed the following strategy: for each dataset, we build the complete graph by linking a sample to every other sample. Then, for each sample $\vec{x}_i$, we compute the approximate distribution of the distances from $\vec{x}_i$ to any other sample $\vec{x}_j$. According to our computational experiments, the most interesting values of ϵ (radius) are obtained by the percentiles p of the distribution belonging to the interval $P = [1, 50]$, but we choose the best model as the one that maximizes the classification accuracy among all values of $p \in [1, 99]$, a process that require the construction of several different graphs for each dataset. Note that unlike the KNN graph, our graph is not regular, in the sense that we allow the degree of the vertices to be quite different. It is worth mentioning that we are using the class labels to choose the best model (model selection), however the feature extraction stage is not supervised, in the sense that the proposed method performs unsupervised metric learning.

Table 1. Average accuracies among KNN, NB, SVM, DT, QDA, MPL, GPC and RFC after dimensionality reduction with PCA, Kernel PCA, PCAKL, GMRF (proposed method), ISOMAP, LLE, Laplacian Eigenmaps, t-SNE and UMAP for several openML.org datasets (2-D case).

	PCA	PCAKL	GMRF	KPCA	ISO	LLE	LAP	tSNE	UMAP
iris	0.941	0.939	**0.978**	0.858	0.906	0.833	0.650	0.957	0.96
prnn_crabs	0.566	0.539	**0.869**	0.578	0.588	0.603	0.558	0.740	0.786
hapiness	0.219	0.494	**0.521**	0.198	0.245	0.182	0.188	0.313	0.245
mux6	0.615	0.591	**0.820**	0.695	0.526	0.630	0.457	0.785	0.636
threeOf9	0.589	0.795	**0.833**	0.532	0.639	0.678	0.588	0.725	0.753
diggle	0.86	0.962	**0.989**	0.897	0.879	0.856	0.784	0.886	0.886
Breast-tissue	0.422	0.610	**0.620**	0.469	0.448	0.492	0.476	0.502	0.469
Hayes-roth	0.581	0.751	**0.772**	0.634	0.638	0.623	0.611	0.666	0.56
AIDS	0.375	0.54	**0.64**	0.35	0.365	0.36	0.37	0.54	0.52
lupus	0.786	0.801	**0.813**	0.758	0.795	0.71	0.704	0.758	0.744
fri_c3_100_5	0.667	0.697	**0.77**	0.667	0.687	0.625	0.668	0.682	0.66
creditscore	0.795	0.843	**0.89**	0.72	0.795	0.76	0.765	0.735	0.78
blogger	0.675	0.75	**0.758**	0.655	0.712	0.712	0.635	0.735	0.645
vineyard	0.793	0.81	**0.817**	0.778	0.793	0.759	0.735	0.788	0.798
veteran	0.668	0.701	**0.741**	0.692	0.643	0.657	0.648	0.708	0.713
Monks	0.587	0.772	**0.880**	0.535	0.557	0.568	0.549	0.649	0.6
xd6	0.682	0.847	**0.858**	0.678	0.728	0.658	0.686	0.78	0.726
corral	0.831	0.909	**1.000**	0.817	0.825	0.675	0.754	0.865	0.796
boxing	0.691	0.695	**0.731**	0.672	0.65	0.654	0.639	0.664	0.645
baskball	0.645	0.609	**0.703**	0.645	0.578	0.604	0.546	0.643	0.614
bolts	0.781	0.917	**0.925**	0.843	0.843	0.675	0.743	0.837	0.818
newton	0.681	0.685	**0.732**	0.690	0.699	0.589	0.596	0.694	0.728
sleuth	0.556	0.565	**0.742**	0.550	0.572	0.604	0.508	0.610	0.584
parity5	0.445	0.546	**0.586**	0.414	0.367	0.351	0.398	0.461	0.367
confidence	0.843	0.865	**0.885**	0.84	0.84	0.802	0.798	0.829	0.85
census6	0.556	0.568	**0.591**	0.552	0.550	0.546	0.550	0.550	0.543
diabetes	0.706	0.715	**0.740**	0.687	0.693	0.664	0.664	0.699	0.689
bodyfat	0.783	0.761	**0.823**	0.702	0.757	0.720	0.637	0.781	0.766
machine_cpu	0.945	0.948	**0.95**	0.935	0.941	0.866	0.844	0.927	0.904
pm10	0.507	0.57	**0.579**	0.529	0.515	0.497	0.508	0.532	0.515
KnuggetChase3	0.764	0.777	**0.808**	0.762	0.773	0.771	0.770	0.768	0.778
fri_c2_100_10	0.697	0.747	**0.787**	0.587	0.650	0.487	0.612	0.650	0.672
Mean	0.664	0.729	**0.788**	0.654	0.662	0.632	0.614	0.702	0.680
Median	0.678	0.749	**0.798**	0.675	0.669	0.656	0.636	0.717	0.701
Minimum	0.219	0.494	**0.558**	0.198	0.245	0.182	0.188	0.313	0.245
Maximum	0.945	0.962	**1.000**	0.935	0.941	0.866	0.844	0.957	0.960
Std. Dev.	0.162	0.136	**0.121**	0.160	0.165	0.149	0.140	0.139	0.155

4 Conclusions and Final Remarks

Unsupervised metric learning and manifold learning are intrinsically related, because, in general, to build robust non-linear features from the observed data, we have to overcome the limitations of the Euclidean metric. In this paper, a generalization of Parametric PCA was proposed to replace the Euclidean distance by the symmetrized KL-divergence between Gaussian-Markov random field models estimated from patches along the ϵ-neighborhood graph. Our claim is that the proposed method is a promising alternative to several dimensionality reduction methods, since the computational experiments supported one main point: the obtained non-linear features can be more discriminative in supervised classification than features obtained by state-of-the-art manifold learning algorithms.

Basically, the good performance of the proposed method can be explained by the incorporation of a patch-based distance function to measure the similarity between the samples, which can be more robust than the pointwise Euclidean distance to deal with the presence of noise and outliers in data. The GMRF model captures the dependence structure of the elements inside a neighborhood system in the graph by means of the inverse temperature parameter and the covariance matrix of the patches. Additionally, unlike autoencoders and other deep-learning based approaches, the proposed method can be used in small sample size problems, without data augmentation techniques, since it does not require large amount of data for convergence.

Future works may include the use of other information-theoretic measures, such as the Fisher information matrix, which is the metric tensor of the parametric space, used to approximate geodesic distances between different statistical models. Different probability density functions can be used to model the features. Many datasets have multi-modal features, that is, their distribution is highly non-Gaussian. Gaussian Mixture Models can be employed to capture this kind of behavior. Generalized Gaussians are also versatile and tractable models for non-Gaussian data. Furthermore, a supervised version of the proposed method can be designed in a way that only neighboring samples of the same class are considered in a patch. Finally, better ways to automatic select the radius parameter in the ϵ-neighborhood graph can be explored, such as the use of different radiuses for each vertex of the graph.

Acknowledgments. This study was financed in part by the Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - Brasil (CAPES) - Finance Code 001.

References

1. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Comput. **15**(6), 1373–1396 (2003)
2. Besag, J.: Spatial interaction and the statistical analysis of lattice systems. J. Royal Stat. Soc. Ser. B (Methodological) **36**(2), 192–236 (1974)
3. Cunningham, J.P., Ghahramani, Z.: Linear dimensionality reduction: survey, insights, and generalizations. J. Mach. Learn. Res. **16**, 2859–2900 (2015)

4. Hammersley, J.M., Clifford, P.: Markov field on finite graphs and lattices (preprint) (1971). www.statslab.cam.ac.uk/~grg/books/hammfest/hamm-cliff.pdf
5. Jolliffe, I.T.: Principal Component Analysis. 2 edn. Springer, New York (2002). https://doi.org/10.1007/b98835
6. Levada, A.L.M.: Parametric PCA for unsupervised metric learning. Pattern Recogn. Lett. **135**, 425–430 (2020)
7. Levada, A.L.M.: Information geometry, simulation and complexity in gaussian random fields. Monte Carlo Methods Appl. **22**, 81–107 (2016)
8. Levada, A.L.M.: PCA-KL: a parametric dimensionality reduction approach for unsupervised metric learning. Adv. Data Anal. Classif. 1–40 (2021). https://doi.org/10.1007/s11634-020-00434-3
9. Li, D., Tian, Y.: Survey and experimental study on metric learning methods. Neural Netw. **105**, 447–462 (2018)
10. McClurkin, J., L.M. Optican, B., Gawne, T.: Concurrent processing and complexity of temporally encoded neuronal messages in visual perception. Science **253**, 675–677 (1991)
11. McInnes, L., Healy, J., Melville, J.: UMAP: Uniform manifold approximation and projection for dimension reduction. arXiv arXiv:1802.03426 (2020)
12. Murase, H., Nayar, S.: Visual learning and recognition of 3d objects from appearance. Int. J. Comput. Vis. **14**, 5–24 (1995)
13. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**, 2323–2326 (2000)
14. Schölkopf, B., Smola, A., Müller, K.-R.: Kernel principal component analysis. In: Gerstner, W., Germond, A., Hasler, M., Nicoud, J.-D. (eds.) ICANN 1997. LNCS, vol. 1327, pp. 583–588. Springer, Heidelberg (1997). https://doi.org/10.1007/BFb0020217
15. Seung, H.S., Lee, D.D.: The manifold ways of perception. Science **290**, 2268–2269 (2000)
16. Tasoulis, S., Pavlidis, N.G., Roos, T.: Nonlinear dimensionality reduction for clustering. Pattern Recog. **107**, 107508 (2020)
17. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**, 2319–2323 (2000)
18. Van Der Maaten, L., Hinton, G.E.: Visualizing high-dimensional data using t-sne. J. Mach. Learn. Res. **9**, 2579–2605 (2008)
19. Van Der Maaten, L., Postma, E., Van den Herik, J.: Dimensionality reduction: a comparative review. J. Mach. Learn. Res. **10**, 66–71 (2009)
20. Wang, F., Sun, J.: Survey on distance metric learning and dimensionality reduction in data mining. Data Min. Knowl. Disc. **29**(2), 534–564 (2014). https://doi.org/10.1007/s10618-014-0356-z