Research Square

Preprints are preliminary reports that have not undergone peer review.
They should not be considered conclusive, used to inform clinical practice,
or referenced by the media as validated information.

Entropic Principal Component Analysis using Cauchy Schwarz divergence

Eduardo Nakao (eduardokazuonakao@gmail.com)
Federal University of São Carlos
Alexandre Levada
Federal University of São Carlos

Short Report

Keywords: Metric learning , Dimensionality reduction , Principal Component Analysis , Information theory , Cauchy-Schwarz divergence

Posted Date: April 1st, 2022

DOI: https://doi.org/10.21203/rs.3.rs-1499062/v1

License: This work is licensed under a Creative Commons Attribution 4.0 International License.
Read Full License

Knowledge and Information Systems manuscript No.
(will be inserted by the editor)

Entropic Principal Component Analysis using Cauchy Schwarz divergence

Eduardo K. Nakao · Alexandre L. M. Levada

Received: date / Accepted: date

Abstract Dimensionality reduction has been explored to address the curse of dimensionality in high dimensional datasets of modern pattern recognition applications. In pattern recognition tasks, it is important to quantify how distinct two data samples are. Unsupervised metric learning serves for this purpose. In dimensionality reduction, a more adequate metric for a given dataset is implicitly learned. Principal Component Analysis is still the most used dimensionality reduction algorithm. Several modifications of this method have already been proposed as other algorithms belonging to the nonlinear class as well. However, all of them somehow rely on the Euclidean norm, which is known to fail in high dimensions and to be sensitive to outliers. So, in this paper, a new entropic approach was proposed, where the neighborhood of a data sample was mapped to an entropic space, where a stochastic divergence replaces the Euclidean. This approach was adopted to compute a new entropic covariance matrix that does not use inner product to estimate correlation between two features. A data sample neighborhood was mapped into an univariate Gaussian distribution and the statistical distance used was the Cauchy-Schwarz divergence. This new matrix was supplied to Principal Component Analysis classic algorithm. We compared the new method with existing linear and nonlinear algorithms. Using several real datasets, the comparison was made under two perspectives: cluster analysis and classification. Using a statistical test, it was possible to conclude that the new approach led to significant better results in both perspectives in comparison to all other algorithms considered.

Keywords Metric learning · Dimensionality reduction · Principal Component Analysis · Information theory · Cauchy-Schwarz divergence

Eduardo K. Nakao
Computing Department, Federal University of São Carlos
Tel.: +55-16-997193190
E-mail: eduardokazuonakao@gmail.com

Alexandre L. M. Levada
Computing Department, Federal University of São Carlos
Tel.: +55-16-33518252
Fax: +55-16-33518252
E-mail: alexandre.levada@ufscar.br

1 Introduction

High dimensional data is present in several domains of science. A huge quantity of features and samples is common on modern pattern recognition and machine learning applications datasets. While a big quantity of examples is good for those tasks, the increase in the feature number can bring negative consequences [6, 8, 12, 18, 21, 30, 36].

The *curse of dimensionality* phenomena states that, as the quantity m of features grows, more samples are needed to approximate the data governing function [9, 13, 26]. Thus, a large sample size n is required in order to extract relevant information from high-dimensional data, but, in real-world contexts, n is limited or even scarce in relation to m. Therefore, a natural way to mitigate this problem is to reduce the data dimensionality m.

Supervised classification in high dimensional spaces can be difficult because Euclidean properties are lost in high dimensions [14, 26]. The Euclidean norm (which is based on the inner product between two vectors) is bigger as the feature quantity grows in a $\sqrt{m}$ proportion. On the other hand, when m is large, the norms variance tends to concentrate around some constant. This is known as the concentration phenomena [18], where it is observed that Euclidean norm loses discrimination power as space dimension raises.

In pattern recognition, the goal is to develop mathematical models for automatic discovery of regularities in data through computational algorithms. In order to extract relevant information from a vast amount of data, one of the key issues in pattern recognition and machine learning is the definition of a suitable similarity measure between samples [20, 32]. Being able to properly quantify how far apart two different observations are, is crucial for any kind of data analysis. In this context, unsupervised metric learning methods try to overcome this issue by finding suitable distance functions for the dataset.

Dimensionality reduction (DR) methods are mathematical tools for data analysis and metric learning. The intuition behind these methods is that, usually, the observed data samples lie along a low-dimensional structure embedded in a high-dimensional input space. The low dimensional space reflects some unknown underlying parameters (i.e., local coordinates) that are encoded in the original feature space. Attempting to uncover this hidden structure in a dataset is the major goal of DR algorithms. It has been shown that these methods have strong relation to metric learning because, besides obtaining a better representation for a given dataset, they also obtain and a distance metric that quantify dissimilarity between its samples in a more appropriate way [3, 20, 32, 28, 33]. Therefore, besides helping data visualization and alleviating the computational burden, these methods also handle the curse of dimensionality by learning an adaptive data-dependent similarity measure that leads to a more compact data representation.

Among all DR methods, Principal Component Analysis (PCA) [16] is still the main algorithm used by researchers. It is based on finding the orthogonal directions that maximize the data variance. This is optimal from a data representation point of view, since it is equivalent to the mean square error minimization between the original and the reduced representation. For this reason, after the PCA transformation, data is organized in clusters with large scattering, which is undesirable for classification problems.

Non-linear DR techniques and PCA also have the limitation of using the L_2 distance, which does not work so well in outlier presence, where classification accuracy decrease can be observed [1]. In high dimensions, there is inconsistency and upward bias in the covariance matrix eigenvectors and eigenvalues [15]. So, the use of traditional covariance matrix to characterize data distribution, may not be a reasonable choice. Some alternatives to this limitation have already been investigated [31].

To overcome this problem, in this paper is proposed a new patch-based approach that maps the KNN graph neighborhoods to an entropic feature space. In this new space, the Euclidean distance between two vectors is replaced by an information-theoretic measure between two statistical models in the covariance matrix construction. In other words, the distance in the feature space is replaced by a statistical divergence between probability distributions defined in the neighborhood of each sample. In this paper we will use the Cauchy-Schwarz divergence [11].

Overall, the obtained results show that the proposed method is capable to improve three major aspects of other DR methods compared: 1) it less sensitive to outliers and noise in data due to its patch-based characteristic; 2) the obtained clusters show a lower intra-class scattering; 3) the extracted features have more discriminant power providing higher supervised classification accuracies.

The paper is divided in the following way: Section 2 shows in details the proposed method for unsupervised metric learning via DR. In Section 3, we detail the experiments, results, and compare several non-linear and linear algorithms with the proposed method. In Section 4, some conclusions are presented. In Section 5, some future work possibilities are discussed.

2 PCA using Cauchy-Schwarz divergence

In the theoretical formulation, we will define the dataset as being the set $X = \{\vec{x}_1, \vec{x}_2, ..., \vec{x}_n\}$, where $\vec{x}_i \in R^m$. If, for all i, $\vec{x}_i$ is linked with its k nearest neighbors, the KNN graph is defined as $G = (V, E)$, where $|V| = n$. The Euclidean distance can be used for this linkage, assuming that a neighborhood is an Euclidean subspace itself [24]. Although, other metrics such as Jaccard, Minkowski and Cosine can be used also. A patch P_i is defined as $\{\vec{x}_i\} \cup \{\vec{x}_j \in N(i)\}$, with $N(i)$ being the neighborhood of $\vec{x}_i$. So

$$P_i = [\vec{x}_i, \vec{x}_{i1}, \vec{x}_{i2}, ..., \vec{x}_{ik}] \tag{1}$$

is the $m \times (k+1)$ matrix that represents the i-th patch. We assume that each row of the matrix P_i is a sample of size $k+1$ of a univariate random variable x, characterized by a probability density function $p(x; \vec{\theta})$, where $\vec{\theta} \in R^L$ is a vector of L parameters. In this study, we consider a Gaussian model, that is, $L = 2$ and $\theta_1 = \mu$ denotes the mean and $\theta_2 = \sigma^2$ denotes the variance. So each random variable corresponds to one of m input features. Each P_i is mapped to a m-dimensional vector of 2D tuples, where each tuple j for $j = 1, ..., m$ has the maximum likelihood estimators of the parameters for each one of the features. In other words, we compute the sample mean and variance of each **line** of the matrix P_i.

The entropic feature vector $\vec{p}_i$ for the patch P_i is given by:

$$\vec{p}_i = \left[\vec{\theta}_1^{(i)}, \vec{\theta}_2^{(i)}, ..., \vec{\theta}_m^{(i)}\right] \tag{2}$$

where each component is a tuple of two parameters:

$$\vec{\theta}_j^{(i)} = \left(\mu_j^{(i)}, (\sigma_j^2)^{(i)}\right) \tag{3}$$

Figure 1 shows the mapping from a patch P_i to an entropic feature vector $\vec{p}_i$.

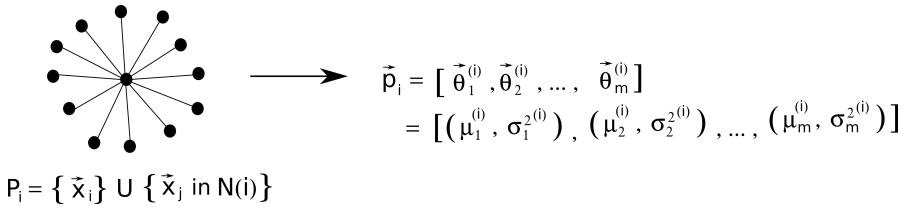

Fig. 1 Mapping from a patch P_i on a graph to an entropic feature vector $\vec{p}_i$

The set of all $\vec{p}_i$, for $i = 1, 2, ..., n$ defines the entropic feature space. We can associate to the entropic feature space, a centroid, which represents the average distribution:

$$\tilde{\vec{p}} = \frac{1}{n}\sum_{i=1}^{n}\vec{p}_i \tag{4}$$

Let the entropic difference between two vectors $\vec{p}_i$ and $\vec{p}_j$ in the entropic feature space be the Cauchy-Schwarz divergence between each one of the tuples in the vectors:

$$\begin{aligned}\vec{p}_i - \vec{p}_j &= \left[D_{CS}(\vec{\theta}_1^{(i)}, \vec{\theta}_1^{(j)}), ..., D_{CS}(\vec{\theta}_m^{(i)}, \vec{\theta}_m^{(j)})\right] \\ &= \vec{d}_{CS}\left(\vec{p}_i, \vec{p}_j\right)\end{aligned} \tag{5}$$

where $D_{CS}(p, q)$ is the Cauchy-Schwarz divergence between probability density functions p and q [11]:

$$\begin{aligned}D_{CS}(p, q) &= -log\frac{\int p(x)q(x)dx}{\sqrt{\int p(x)^2dx \int q(x)^2dx}} \\ &= \frac{1}{2}log\left(\int p(x)^2dx\right) + \frac{1}{2}log\left(\int q(x)^2dx\right) - log\left(\int p(x)q(x)dx\right)\end{aligned} \tag{6}$$

In this study, we assume a univariate Gaussian model for each feature, that is, we have the distributions $p(x|\vec{\theta}_i)$ and $q(x|\vec{\theta}_j)$ as $N(\mu_1, \sigma_1^2)$ and $N(\mu_2, \sigma_2^2)$, respectively, which leads to [27]:

$$D_{CS}(p, q) = \frac{1}{2}log\left(\frac{(\sigma_1^2 + \sigma_2^2)^2}{4\sigma_1^2\sigma_2^2}\right) + \frac{(\mu_1 - \mu_2)^2}{\sigma_1^2 + \sigma_2^2} \tag{7}$$

We define the entropic kernel matrix C as a surrogate for the covariance matrix:

$$C = \frac{1}{n-1} \sum_{i=1}^{n} \vec{d}_{CS}(\vec{p}_i, \tilde{\vec{p}}) \vec{d}_{CS}(\vec{p}_i, \tilde{\vec{p}})^T \tag{8}$$

where $\vec{d}_{CS}(\vec{p}_i, \tilde{\vec{p}})$ is a m-dimensional vector of Cauchy-Schwarz divergences. So an *entropic covariance matrix*, is defined by the relative entropy (i.e., the Kullback-Leibler divergence) between the local distributions estimated from each patch and the average distribution. The Cauchy-Schwarz divergence is equivalent to the Kullback-Leibler divergence for the quadratic entropy.

The following schemes illustrate the procedure from the feature covariance computation point of view. In Figure 2 is represented a space which its span basis is formed by the dataset samples vectors. Each vector dispersed in this space corresponds to a feature vector f_i. So the ith coordinate of f_i is the value that $\vec{x}_1$ has in the ith feature. In blue we can see that, originally, the covariance between f_i and f_j involves the inner product of these two vectors. Thus, the covariance can also be related to the Euclidean distance between them.

In Figure 3 is represented the transformation to the entropic feature space, that is, the mapping between a KNN graph to an univariate Gaussian distribution. In the entropic space, the covariance between f_i and f_j no longer involves the inner product between its vectors, but instead, the Cauchy-Schwarz divergence between distributions p and q.

Notice that, all the proposed procedure does, is obtaining a new covariance matrix that does not involve inner products, where a contextual approach replaces the pointwise metric. Therefore, this new matrix can be used in the default PCA algorithm. The final PCA projection matrix, responsible for the linear projection from old to new coordinates, can be normally built with the new entropic covariance matrix eigenvectors. So, from now on, we will refer the PCA method that uses the entropic covariance matrix as *Cauchy-Schwarz PCA* (CSPCA). As a final remark, it is important to highlight also that, the use of a projection matrix allows an easy finding of an instance new coordinates in the lower dimensional space, which is a big performance advantage in comparison to manifold learning algorithms.

3 Experiments and results

We compared the proposed method performance against: the original PCA, Joint Sparse PCA [35], Kernel PCA [25], Robust PCA [34], LLE [24], ISOMAP [29] and Laplacian Eigenmaps [2] in several datasets available in `www.openml.org`. Those datasets are very heterogeneous, having significant differences in the number of features (m), samples (n) and classes (c). The experimental analysis is divided in two sets: one focused on internal cluster assessment and another based on classification accuracy.

In the first experimental set, the goal is to assess the quality of the clusters obtained after feature extraction. A cluster is a set of samples that belongs to the same class in the original dataset, given that all samples were labeled in all

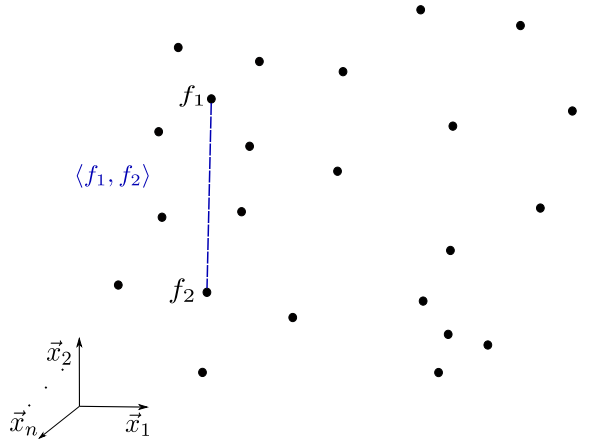

Fig. 2 The sample space representation

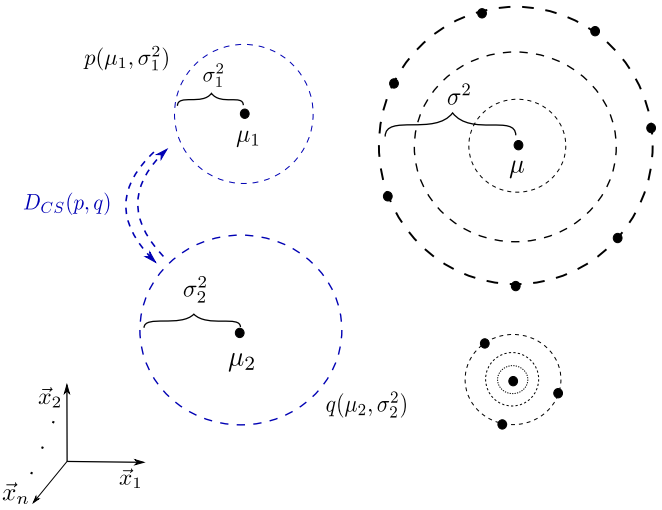

Fig. 3 The entropic space representation

datasets (i.e. every sample belongs to some class). We used the Silhouette Coefficient (SC) [23] to measure the similarity between a given data sample and its own cluster (cohesion) in comparison to different clusters (separation). This measure provides a quantitative way to analyze the consistency within clusters. The idea is to measure, for all clusters, how tight the cluster is. A high SC indicates low intra-class scattering. We can find the results for 30 datasets in Table 1, where column CSPCA denote the proposed entropic method under Gaussian hypothesis. The best result in a line is boldfaced and the second best is underlined. At the bottom of the table are also shown for each feature extraction algorithm the SC average, standard deviation, median and mean absolute deviation (MAD).

The results indicate that, for these datasets, Cauchy-Schwarz PCA builds a more meaningful representation from within clusters consistency perspective than the other methods. Moreover, note that in 26 of 30 datasets, CSPCA obtained the

Table 1 Silhouette coefficients for clusters produced by PCA, Kernel PCA, ISOMAP, LLE, Laplacian Eigenmaps, Joint Sparse PCA, Robust PCA and Cauchy-Schwarz PCA.

	PCA	KPCA	ISO	LLE	LAP	JSPCA	RPCA	CSPCA
iris	0.401	0.469	0.452	0.365	0.541	0.470	0.551	**0.603**
blood	0.086	0.026	0.082	0.008	0.004	0.092	0.083	**0.174**
kc1	0.371	0.210	0.187	0.187	-0.459	0.370	0.369	**0.467**
Australian	0.279	0.276	0.291	0.130	0.346	0.272	0.312	**0.423**
transplant	0.485	0.436	0.486	0.410	0.438	0.480	0.520	**0.542**
servo	0.121	0.105	0.114	0.104	0.085	0.120	**0.279**	0.215
analcatdata	0.151	0.081	0.125	0.149	0.028	0.170	0.107	**0.198**
datatrieve	0.239	0.011	0.096	0.066	0.081	0.236	0.174	**0.264**
machine_cpu	0.498	0.399	0.492	0.496	0.410	0.494	**0.575**	0.508
arsenic-female	0.122	0.008	0.170	0.143	0.030	0.104	0.068	**0.212**
page-blocks	0.419	0.218	0.527	0.581	0.436	0.426	0.419	**0.634**
arsenic-male	0.563	-0.182	0.674	0.697	-0.057	0.504	0.057	**0.731**
mw1	0.349	0.122	0.286	0.175	0.18	0.337	0.346	**0.424**
car	0.029	**0.189**	0.046	0.163	0.079	0.01	0.068	0.182
ar1	0.265	0.028	0.216	-0.004	-0.002	0.276	0.246	**0.437**
diggle_table	0.406	0.409	0.450	0.328	0.304	0.407	0.444	**0.471**
rmftsa_ladata	0.228	0.242	0.238	0.185	0.230	0.225	0.236	**0.296**
kc3	0.386	0.103	0.233	0.045	-0.129	0.394	0.394	**0.569**
diabetes	**0.117**	0.100	0.115	0.101	0.054	0.111	0.106	0.115
mammography	0.349	0.032	0.307	0.070	-0.251	0.348	0.349	**0.640**
bank-marketing	0.082	-0.006	-0.001	0.078	-0.257	0.082	0.082	**0.317**
heart-h	0.056	0.041	0.076	0.087	-0.004	0.066	0.134	**0.205**
molecular	0.106	0.134	0.138	0.035	0.137	0.105	0.170	**0.248**
delta_ailerons	0.117	0.341	0.383	0.077	0.419	0.114	0.117	**0.469**
pc3	0.201	0.074	-0.017	-0.003	-0.341	0.201	0.188	**0.227**
ar4	0.357	0.176	0.318	0.203	0.131	0.361	0.356	**0.473**
KnuggetChase3	0.199	0.070	0.187	0.077	0.091	0.196	0.203	**0.317**
threeOf9	0.034	0.017	0.049	0.095	0.044	0.048	0.029	**0.193**
galaxy	0.179	0.255	0.193	0.235	0.270	0.177	0.219	**0.275**
thoracic_surgery	0.006	-0.002	-0.006	0.082	-0.021	0.008	-0.075	**0.303**
Ave.	0.240	0.146	0.230	0.179	0.094	0.238	0.238	**0.371**
S. Dev.	0.156	0.154	0.178	0.174	0.240	**0.153**	0.167	0.167
Median	0.215	0.104	0.190	0.117	0.080	0.213	0.211	**0.317**
MAD	0.134	**0.124**	0.143	0.127	0.181	0.132	0.138	0.146

highest SC, that is, in 87% of the cases, the method produced better clusters than the others, which indicates that it can be a promising alternative to unsupervised metric learning via DR. Wilcoxon signed-rank test, for a significance level $\alpha = 1\%$, shows that, CSPCA produced significantly better clusters than PCA (p-value $= 1.91 \times 10^{-6}$), Kernel PCA (p-value $= 1.92 \times 10^{-6}$), ISOMAP (p-value $= 2.56 \times 10^{-6}$), LLE (p-value $= 1.73 \times 10^{-6}$), Laplacian Eigenmaps (p-value $= 1.73 \times 10^{-6}$), JSPCA (p-value $= 1.73 \times 10^{-6}$) and Robust PCA (p-value $= 9.31 \times 10^{-6}$).

In the second experimental set, we analyse supervised classification performance. For this purpose, eight different non-parametric and parametric classifiers were used: K-Nearest Neighbors (KNN), Naive Bayes (NB), linear Support Vector Machine (SVM), Decision Trees (DT), Multi-layer Perceptron (MPL), Quadratic Discriminant Analysis (QDA) under Gaussian hypothesis, Random Forest Classifier (RFC) and Gaussian Process Classifier (GPC). In all experiments, we selected 40% of the samples for testing and 60% for training. In some datasets, the QDA classifier was not able to produce results, since there were classes with a single sample, which makes unfeasible the class covariance matrix estimation. Table 2, shows the classification accuracies average of all eight classifiers used for several datasets after the feature extraction processes.

Table 2 Average accuracies in supervised classification by different classifiers after PCA, ISOMAP, Kernel PCA, LLE, Joint Sparse PCA, Laplacian Eigenmaps, Robust PCA and Cauchy-Schwarz PCA.

	PCA	KPCA	ISO	LLE	LAP	JSPCA	RPCA	CSPCA
iris	0.94	0.86	0.91	0.83	0.65	0.94	0.96	**0.98**
engine1	0.81	0.84	0.86	0.73	0.78	0.81	0.86	**0.92**
crabs	0.57	0.58	0.59	0.60	0.56	0.57	0.61	**0.65**
hapiness	0.22	0.20	0.25	0.18	0.19	0.23	0.27	**0.53**
mux6	0.62	0.70	0.53	0.63	0.46	0.64	0.62	**0.83**
threeOf9	0.59	0.53	0.64	0.68	0.59	0.64	0.75	**0.83**
sa_heart	0.65	0.68	0.70	0.67	0.66	0.64	0.69	**0.73**
breast-tissue	0.43	0.49	0.44	0.50	0.51	0.40	0.48	**0.63**
vertebra_column	0.63	0.62	0.67	0.64	0.63	0.65	0.63	**0.76**
transplant	0.98	0.94	0.98	0.93	0.87	**0.99**	0.93	**0.99**
Hayes	0.59	0.63	0.61	0.62	0.60	0.56	0.65	**0.77**
plasma_retinol	0.51	0.56	0.58	0.53	0.58	0.53	0.53	**0.61**
visualizing_livestock	0.29	0.19	0.30	0.28	0.16	0.30	0.20	**0.36**
strikes	0.59	0.59	0.61	0.57	0.56	0.60	0.57	**0.69**
pwLinear(2)	0.64	0.65	0.70	0.70	0.59	0.66	0.79	**0.82**
paraty5	0.46	0.25	0.35	0.43	0.39	0.41	0.44	**0.57**
fruitfly	0.53	0.53	0.49	0.59	0.54	0.49	0.59	**0.65**
AIDS	0.33	0.31	0.33	0.31	0.33	0.34	0.53	**0.59**
lupus	0.79	0.71	0.79	0.70	0.67	0.80	0.66	**0.81**
pm10	0.52	0.52	0.50	0.48	0.50	0.53	0.53	**0.56**
Avg.	0.59	0.57	0.59	0.59	0.54	0.59	0.62	**0.72**
S. Dev.	0.20	0.22	0.21	0.20	0.20	0.20	0.20	**0.17**
Median	0.59	0.58	0.61	0.57	0.56	0.60	0.62	**0.74**
MAD	0.15	0.16	0.17	0.16	0.15	0.16	0.16	**0.14**

The results indicate that the proposed method in average outperformed all other DR methods for these datasets. Wilcoxon signed-rank test, for a significance level of 1%, shows that, CSPCA produced higher classification accuracies than PCA (p-value $= 2.56 \times 10^{-23}$), Kernel PCA (p-value $= 2.66 \times 10^{-27}$), ISOMAP (p-value $= 2.53 \times 10^{-24}$), LLE (p-value $= 7.06 \times 10^{-24}$), Laplacian Eigenmaps (p-value $= 6.32 \times 10^{-27}$), JSPCA (p-value $= 1.36 \times 10^{-23}$) and RPCA (p-value $= 2.46 \times 10^{-21}$).

Target dimensionality used for DR was always equals to 2. This dimensionality allows data dispersion visualization to check DR methods difference. It is well known that the target dimensionality has big influence in the feature extraction step. Several methods for intrinsic dimensionality discovery of some dataset exist [5, 7, 10, 22]. Future works may experiment with some of those discovery strategies such as exhaustive search guided by performance using SC and accuracy. There are also strategies guided by representation power, such as "DR method transformation matrix first largest eigenvalues sum over all eigenvalues sum" analysis [5]. In our work, we fixed the same dimensionality for all DR methods in order to use a fair and simple criteria as for now.

Regarding parameter tuning, it is worth mentioning also that, differently than manifold learning methods, the definition of the patch-neighborhood size K plays an important role in the proposed method. Different values of K can lead to significantly different results. In our experiments, we adopted a supervised linear search to estimate the best value of K for a given dataset. Basically, we defined the set of possible values of K by considering an initial value, and an increment window based on the number of samples n. Then, we computed the average classification accuracies (considering the classifiers previously defined) for several values of K,

and selected the value that maximizes the average accuracy. An intuition behind this choice is that, a smaller K is usually preferred in small datasets to preserve patch locality. But, for suitable parameter estimation, the trade-off between a large enough sample size and locality preservation, must be considered. Also, the best K for supervised classification may not be the best for clustering analysis (for the former purpose, the $\sqrt{n}$ criteria is usual).

4 Conclusions

Results with several real datasets indicated that, besides improving the produced clusters quality, the proposed method can also improve the supervised classification accuracy in comparison to other algorithms. So, in unsupervised metric learning tasks, the use of this new approach can be a better choice than original PCA and some of its modifications, and even manifold learning techniques.

In comparison with the other DR methods considered in this study, the main positive points of the proposed method can be summarized as: 1) It is fast because it does not involve optimization step in its process. 2) In general, the obtained clusters show a lower intra-class scattering, which is interesting for unsupervised classification. 3) It is a patch-based approach (in contrast to PCA, Kernel PCA and other variants that are point-wise methods), which makes it less sensitive to outliers, noise and perturbations in data. 4) In several real datasets, the extracted features provided higher supervised classification accuracies than the other algorithms compared (i.e., the features obtained show in average more discriminant power). 5) Evaluation of new instances is straightforward since, once the projection matrix is built, the mapping is direct; so, unlike manifold learning algorithms, once a new sample arises, there is no need to retrain the model.

Recently, deep learning has been considered by many practitioners and researchers as the state-of-the-art for crafting features from high dimensional datasets, especially from image data [4]. Deep learning is a class of neural networks that uses multiple layers to progressively extract higher level features from the raw input [17]. One requirement for deep learning is to have a large sample size, that is, a huge amount of data is needed in order to properly adjust its parameters. But, this requirement is not always met. DR algorithms on the other hand, are able to learn features from smaller datasets, producing good results even when the number of samples n is less than or equal the number of original features m. Moreover, most deep learning models work with the supervised learning paradigm, since they are generalizations of multi-layer perceptrons, which means that information about the class labels is required, which is not always possible too.

5 Future works

Our results could be further improved by searching in a wider range of values for the parameter K (patch-neighborhood size). In this study, the value of K is global, but we intend to perform analysis of the local Hessian matrix in order to bring insights about how to adjust the K parameter adaptively - samples in areas with lower curvature should have larger neighborhoods and samples in higher curvature

areas could have a smaller neighborhood. A possible problem is the computational cost, since the number of operations in the algorithm would significantly increase

Also, other metrics than Euclidean can be employed in the KNN graph construction. Another possible improvement is a supervised version of CSPCA, which considers only neighbors that belong to the same class of the central data sample.

The same new entropic approach proposed for PCA in this paper could be also incorporated in Linear Discriminant Analysis and Isometric Feature Mapping methods for example. Extensions to non-linear DR by the incorporation of different kernels can be considered.

The proposed method can be extended to other statistical divergences and models. It is straightforward to generalize the method to distinct probability densities. The Cauchy-Schwarz divergence can be calculated to other distributions. If the dataset has multi-modal features, Gaussian Mixture Model can be used. Kernel Density Estimation is another possibility to this modeling.

We were able to show already that the method is efficient with the Bhattacharyya distance as well [19], which is a sign of its robustness. So future works can experiment other information-theoretic measures such as Renyi, Sharma-Mittal, Tsallis and Total Variation.

Regarding the performance evaluation steps, other metrics (e.g., Adjusted Rand Index, Kappa) and tests (e.g., Friedman) can be used.

Conflict of interest

The authors declare that they have no conflict of interest.

References

1. Abboud, M., Benzinou, A., Nasreddine, K.: A robust tangent pca via shape restoration for shape variability analysis. Pattern Analysis and Applications (2019)
2. Belkin, M., Niyogi, P.: Laplacian eigenmaps for dimensionality reduction and data representation. Neural Computation **15**(6), 1373–1396 (2003)
3. Bellet, A., Habrard, A., Sebban, M.: A survey on metric learning for feature vectors and structured data (2013)
4. Bengio, Y., Courville, A., Vincent, P.: Representation learning: A review and new perspectives. IEEE Transactions on Pattern Analysis and Machine Intelligence **35**(8), 1798–1828 (2013)
5. Camastra, F.: Data dimensionality estimation methods: a survey. Pattern Recognition **36**(12), 2945 – 2954 (2003). DOI https://doi.org/10.1016/S0031-3203(03)00176-6. URL `http://www.sciencedirect.com/science/article/pii/S0031320303001766`
6. Chávez, E., Navarro, G., Baeza-Yates, R., Marroquín, J.L.: Searching in metric spaces. ACM Computing Surveys **33**(3), 273–321 (2001)
7. Cox, T.F., Cox, M.A.A.: Multidimensional Scaling, *Monographs on Statistics and Applied Probability*, vol. 88. Chapman & Hall (2001)
8. Debie, E., Shafi, K.: Implications of the curse of dimensionality for supervised learning classifier systems: theoretical and empirical analyses. Pattern Analysis and Applications **22**, 519–536 (2019)
9. Fukunaga, K.: Introduction to Statistical Pattern Recognition. Academic Press (1990)
10. He, J., Ding, L., Jiang, L., Li, Z., Hu, Q.: Intrinsic dimensionality estimation based on manifold assumption. Journal of Visual Communication and Image Representation **25**(5), 740 – 747 (2014). DOI https://doi.org/10.1016/j.jvcir.2014.01.006. URL `http://www.sciencedirect.com/science/article/pii/S1047320314000078`

11. Hoang, H.G., Vo, B.N., Vo, B.T., Mahler, R.: The cauchy-schwarz divergence for poisson point processes. IEEE Trans. on Information Theory **61**(8), 4475–4485 (2015)
12. Hughes, G.F.: On the mean accuracy of statistical pattern recognizers. IEEE Transactions on Information Theory **14**(1), 55–63 (1968)
13. Hwang, J., Lay, S., Lippman, A.: Nonparametric multivariate density estimation: A comparative study. IEEE Trans. on Signal Processing **42**(10), 2795–2810 (1994)
14. Jimenes, L.O., Landgrebe, D.: Supervised classification in high dimensional space: Geometrical, statistical and asymptotical properties of multivariate data. IEEE Trans. on Syst., Man and Cybern. **28**(1), 39–54 (1998)
15. Johnstone, I.M., Paul, D.: Pca in high dimensions: An orientation. Proc. of the IEEE **106**(8), 1277–1292 (2018)
16. Jolliffe, I.T.: Principal Component Analysis, 2 edn. Springer (2002)
17. LeCun, Y., Bengio, Y., Hinton, G.: Deep learning. Nature **521**, 436–444 (2015)
18. Lee, J.A., Verleysen, M.: Nonlinear Dimensionality Reduction. Springer (2007)
19. Levada, A.L.: Parametric pca for unsupervised metric learning. Pattern Recognition Letters **135**, 425–430 (2020)
20. Li, D., Tian, Y.: Survey and experimental study on metric learning methods. Neural Networks **105**, 447–462 (2018)
21. Marimont, R., Shapiro, M.: Nearest neighbour searches and the curse of dimensionality. IMA Journal of Applied Mathematics **24**(1), 59–70 (1979)
22. Miranda, G.F., Thomaz, C.E., Giraldi, G.A.: Geometric data analysis based on manifold learning with applications for image understanding. In: 2017 30th SIBGRAPI Conference on Graphics, Patterns and Images Tutorials (SIBGRAPI-T), pp. 42–62 (2017). DOI 10.1109/SIBGRAPI-T.2017.9
23. Rousseeuw, P.J.: Silhouettes: A graphical aid to the interpretation and validation of cluster analysis. Journal of Comp. and Appl. Math. **20**, 53–65 (1987)
24. Roweis, S., Saul, L.: Nonlinear dimensionality reduction by locally linear embedding. Science **290**, 2323–2326 (2000)
25. Schölkopf, B., Smola, A., Müller, K.R.: Kernel principal component analysis. In: Advances in Kernel Methods – Support Vector Learning, pp. 327–352. MIT Press (1999)
26. Scott, D.W.: Multivariate Density Estimation. John Wiley & Sons (1992)
27. Spurek, P., Palka, W.: Clustering of gaussian distributions. In: 2016 International Joint Conference on Neural Networks (IJCNN), pp. 3346–3353 (2016)
28. Suárez, J.L., García, S., Herrera, F.: A tutorial on distance metric learning: Mathematical foundations, algorithms and software (2018)
29. Tenenbaum, J.B., de Silva, V., Langford, J.C.: A global geometric framework for nonlinear dimensionality reduction. Science **290**, 2319–2323 (2000)
30. Trunk, G.V.: A problem of dimensionality: A simple example. IEEE Trans. on Pattern Analysis and Machine Intelligence **1**(3), 306–307 (1979)
31. Vaswani, N., Chi, Y., Bouwmans, T.: Rethinking pca for modern data sets: Theory, algorithms, and applications. Proc. of the IEEE **106**(8), 1274–1276 (2018)
32. Wang, F., Sun, J.: Survey on distance metric learning and dimensionality reduction in data mining. Data Min. Knowl. Discov. **29**(2), 534–564 (2015)
33. Yang, L., Jin, R.: Distance metric learning: A comprehensive survey. Michigan State Universiy (2006)
34. Yang, T.N., Wang, S.D.: Robust algorithms for principal component analysis. Pattern Recognition Letters **20**(9), 927–933 (1999)
35. Yi, S., Lai, Z., He, Z., ming Cheung, Y., Liu, Y.: Joint sparse principal component analysis. Pattern Recognition **61**, 524 – 536 (2017)
36. Zimek, A., Schubert, E., Kriegel, H.P.: A survey on unsupervised outlier detection in high-dimensional numerical data. Statistical Analysis and Data Mining **5**(5), 363–387 (2012)