Modified Divergences for Gaussian Densities

Karim T. Abou–Moustafa[1] and Frank P. Ferrie[2]

[1] Robotics Institute, Carnegie Mellon University
5000 Forbes Avenue, Pittsburgh, PA 15213, U.S.A
karimt@andrew.cmu.edu
[2] Dept. of Electrical & Computer Engineering, McGill University
3480 University Street, Montréal, QC, H3A 0E9, Canada
ferrie@cim.mcgill.ca

Abstract. Multivariate Gaussian densities are pervasive in pattern recognition and machine learning. A central operation that appears in most of these areas is to measure the difference between two multivariate Gaussians. Unfortunately, traditional measures based on the Kullback–Leibler (KL) divergence and the Bhattacharyya distance do not satisfy all metric axioms necessary for many algorithms. In this paper we propose a modification for the KL divergence and the Bhattacharyya distance, for multivariate Gaussian densities, that transforms the two measures into distance metrics. Next, we show how these metric axioms impact the unfolding process of manifold learning algorithms. Finally, we illustrate the efficacy of the proposed metrics on two different manifold learning algorithms when used for motion clustering in video data. Our results show that, in this particular application, the new proposed metrics lead to significant boosts in performance (at least 7%) when compared to other divergence measures.

1 Introduction

There are various applications in machine learning and pattern recognition in which the data of interest $\mathcal{D}$ are represented as a family or a collection of sets $\mathcal{D} = \{\mathcal{S}_i\}_{i=1}^{n}$, where $\mathcal{S}_i = \{\mathbf{x}_j^i\}_{j=1}^{n_i}$, and $\mathbf{x}_j^i \in \mathbb{R}^p$. For some of these applications, it is reasonable to model each $\mathcal{S}_i$ as a Gaussian distribution $\mathcal{G}_i(\boldsymbol{\mu}_i, \boldsymbol{\Sigma}_i)$ with mean vector $\boldsymbol{\mu}_i$ and a covariance matrix $\boldsymbol{\Sigma}_i$.[3] In these settings, a natural measure for the (dis)similarity between two Gaussians, $\mathcal{G}_1$ and $\mathcal{G}_2$ say, is the divergence measure of probability distributions [3, 6]. For instance, some of the well known divergence measures with closed form expressions for Gaussian densities are the symmetric Kullback–Leibler (KL), or Jeffreys, divergence $d_J(\mathcal{G}_1, \mathcal{G}_2)$ [12], the Bhattacharyya distance $d_B(\mathcal{G}_1, \mathcal{G}_2)$ and the Hellinger distance $d_H(\mathcal{G}_1, \mathcal{G}_2)$ [9].

When considering a learning problem such as classification, clustering, or low dimensional embedding for the family of sets $\mathcal{D}$, via its representation as the set

[3] Notations: Bold small letters $\mathbf{x}, \mathbf{y}$ are vectors. Bold capital letters $\mathbf{A}, \mathbf{B}$ are matrices. Calligraphic and double bold capital letters $\mathcal{X}$, $\mathcal{Y}$, $\mathbb{X}$, $\mathbb{Y}$ denote sets and/or spaces. Positive (semi-)definite matrices, PD (and PSD) are denoted by $\mathbf{A} \succ 0$ and $\mathbf{A} \succeq 0$ respectively. $\mathrm{tr}(\cdot)$ is the matrix trace. $|\cdot|$ is the matrix determinant. $\mathbf{I}$ is the identity matrix.

of Gaussians $\{\mathcal{G}_i\}_{i=1}^n$, a natural question that arises is that of *which divergence measure will yield a better performance?* At first glance, one can consider an answer along two main dimensions: 1) the learning algorithm that shall be used for the sought task, and 2) the data set under consideration. In this research, however, we show that the metric properties of these divergence measures form a third crucial dimension that has a direct impact on the algorithm's performance. In particular, we show that when modifying the closed form expressions for $d_J(\mathcal{G}_1, \mathcal{G}_2)$ and $d_B(\mathcal{G}_1, \mathcal{G}_2)$ such that both measures satisfy all metric axioms[4], the resulting new measures yield significant improvements in the discriminability of the embedding spaces obtained from two different manifold learning algorithms, classical Multidimensional Scaling (cMDS) [21] and Laplacian Eigenmaps (LEM) [4]. These improvements in discriminability, in turn, result in consistent and significant boosts in clustering accuracy. For the application considered in this paper, motion clustering in video data, an improvement in discriminability of at least 7% is observed.

Starting with the closed form expressions for $d_J(\mathcal{G}_1, \mathcal{G}_2)$ and $d_B(\mathcal{G}_1, \mathcal{G}_2)$, in Section (2) we take, a closer look on how each term in these expressions violate the desired metric axioms. Then, we propose modifications for these expressions that result in new distances that satisfy all metric axioms. In Section (3), we show how the metric properties of divergence can impact the unfolding process of manifold learning algorithms such as cMDS and LEM. In Section (4), we compare the performance of cMDS and LEM using the proposed divergence measures in the context of clustering human motion in video data. Finally, conclusions are drawn in Section (5).

2 Characteristics of $d_J(\mathcal{G}_1, \mathcal{G}_2)$ & $d_B(\mathcal{G}_1, \mathcal{G}_2)$

Our discussion begins with the characteristics of $d_J(\mathcal{G}_1, \mathcal{G}_2)$ and $d_B(\mathcal{G}_1, \mathcal{G}_2)$ in terms of structure and metric properties. Let $\mathbb{G}_p$ be the family of p–dimensional Gaussian densities, where the density $\mathcal{G}(\boldsymbol{\mu}, \boldsymbol{\Sigma}) \in \mathbb{G}_p$ is defined as:

$$\mathcal{G}(\mathbf{x}; \boldsymbol{\mu}, \boldsymbol{\Sigma}) = (2\pi)^{-\frac{p}{2}} |\boldsymbol{\Sigma}|^{-\frac{1}{2}} \exp\{-\tfrac{1}{2}(\mathbf{x} - \boldsymbol{\mu})^\top \boldsymbol{\Sigma}^{-1} (\mathbf{x} - \boldsymbol{\mu})\},$$

$\mathbf{x}, \boldsymbol{\mu} \in \mathbb{R}^p$, $\boldsymbol{\Sigma} \in \mathbb{S}_{++}^{p \times p}$, and $\mathbb{S}_{++}^{p \times p}$ is the manifold of symmetric positive definite (PD) matrices. For $\mathcal{G}_1, \mathcal{G}_2 \in \mathbb{G}_p$, Jeffreys divergence has the closed form expression:

$$d_J(\mathcal{G}_1, \mathcal{G}_2) = \tfrac{1}{2}\mathbf{u}^\top \boldsymbol{\Psi} \mathbf{u} + \tfrac{1}{2}\mathrm{tr}\{\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_2 + \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1 - 2\mathbf{I}\}, \tag{1}$$

where $\boldsymbol{\Psi} = (\boldsymbol{\Sigma}_1^{-1} + \boldsymbol{\Sigma}_2^{-1})$, and $\mathbf{u} = (\boldsymbol{\mu}_1 - \boldsymbol{\mu}_2)$. The Bhattacharyya coefficient ρ, which is a measure of similarity between probability distributions, is defined as:

$$\rho(\mathcal{G}_1, \mathcal{G}_2) = |\boldsymbol{\Gamma}|^{-\frac{1}{2}} |\boldsymbol{\Sigma}_1|^{\frac{1}{4}} |\boldsymbol{\Sigma}_2|^{\frac{1}{4}} \exp\{-\tfrac{1}{8}\mathbf{u}^\top \boldsymbol{\Gamma}^{-1} \mathbf{u}\}, \tag{2}$$

[4] A metric space [11, p. 3] is an ordered pair $(\mathcal{X}, d)$, where $\mathcal{X}$ is a non-empty abstract set (of any objects/elements whose nature is left unspecified), and d is a distance function, or a metric, defined as: $d : \mathcal{X} \times \mathcal{X} \mapsto \mathbb{R}$, and $\forall\ a, b, c \in \mathcal{X}$, the following axioms hold : (i) $d(a, b) \geq 0$, (ii) $d(a, a) = 0$, (iii) $d(a, b) = 0$ iff $a = b$, (iv) Symmetry : $d(a, b) = d(b, a)$, and (v) The triangle inequality : $d(a, c) \leq d(a, b) + d(b, c)$. Semi-metrics satisfy axioms (i), (ii), and (iv) only. Note that the axiomatic definition of metrics and semi-metrics, in particular axioms (i) and (ii), produce the positive semi-definiteness of d. Hence metrics and semi-metrics are PSD.

where $\boldsymbol{\Gamma} = (\frac{1}{2}\boldsymbol{\Sigma}_1 + \frac{1}{2}\boldsymbol{\Sigma}_2)$. From $\rho(\mathcal{G}_1, \mathcal{G}_2)$, the Hellinger distance d_H is defined as $\sqrt{2[1 - \rho(\mathcal{G}_1, \mathcal{G}_2)]}$, while the Bhattacharyya distance d_B is $-\log \rho(\mathcal{G}_1, \mathcal{G}_2)$, which also yields an interesting closed form expression:

$$d_B(\mathcal{G}_1, \mathcal{G}_2) = \tfrac{1}{8}\mathbf{u}^\top \boldsymbol{\Gamma}^{-1}\mathbf{u} + \tfrac{1}{2}\ln\left\{|\boldsymbol{\Sigma}_1|^{-\frac{1}{2}}|\boldsymbol{\Sigma}_2|^{-\frac{1}{2}}|\boldsymbol{\Gamma}|\right\}. \tag{3}$$

It is well known that the KL divergence is not a metric since it does not satisfy the triangle inequality [12], and hence d_J in (1) is not a metric. Similarly, d_B in (3) is not a metric for the same reason, however, d_H is indeed a metric [9].

The two closed form expressions in Equations (1) and (3) have the same structure which is a summation of two components in terms of their first and second order moments. The first term in Equations (1) and (3) measures the difference between the means $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$ weighted by the covariance matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. The second term, however, measures the difference or discrepancy between the covariance matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ only, and is independent from the means $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$.

The first term in Equations (1) and (3), up to a scale factor and a square root, is equivalent to the generalized quadratic distance (GQD) between $\mathbf{x}, \mathbf{y} \in \mathbb{R}^p$: $d(\mathbf{x}, \mathbf{y}; \mathbf{A}) = \sqrt{(\mathbf{x} - \mathbf{y})^\top \mathbf{A}(\mathbf{x} - \mathbf{y})}$, where $\mathbf{A} \in \mathbb{S}_{++}^{p \times p}$. If $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \mathbf{\Sigma}$, then Equations (1) and (3) reduce to:

$$d_J(\mathcal{G}_1, \mathcal{G}_2) = \tfrac{1}{2}\mathbf{u}^\top \boldsymbol{\Psi}\mathbf{u}, \text{ and} \tag{4}$$

$$d_B(\mathcal{G}_1, \mathcal{G}_2) = \tfrac{1}{8}\mathbf{u}^\top \boldsymbol{\Gamma}^{-1}\mathbf{u}. \tag{5}$$

Note that the squared GQD $d^2(\mathbf{x}, \mathbf{y}; \mathbf{A})$ is a semi-metric, and if $\mathbf{A}$ is PSD, then $d(\mathbf{x}, \mathbf{y}; \mathbf{A})$ is a pseudo-metric. Both, semi-metrics and pseudo metrics, do not satisfy the triangle inequality, and hence Equations (4) and (5) are semi-metrics. Further, if $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \mathbf{I}$, then Equations (4) and (5), up to a scale factor, reduce to the squared Euclidean distance. Note that the squared Euclidean distance is also a semi-metric. The second term in Equations (1) and (3) is the distance or discrepancy measure between $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$, and is independent of $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$. If $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \boldsymbol{\mu}$ then:

$$d_J(\mathcal{G}_1, \mathcal{G}_2) = \tfrac{1}{2}\mathrm{tr}\{\boldsymbol{\Sigma}_1^{-1}\boldsymbol{\Sigma}_2 + \boldsymbol{\Sigma}_2^{-1}\boldsymbol{\Sigma}_1 - 2\mathbf{I}\}, \text{ and} \tag{6}$$

$$d_B(\mathcal{G}_1, \mathcal{G}_2) = \tfrac{1}{2}\ln\left\{|\boldsymbol{\Gamma}||\boldsymbol{\Sigma}_1|^{-\frac{1}{2}}|\boldsymbol{\Sigma}_2|^{-\frac{1}{2}}\right\}. \tag{7}$$

Since Equations (1) and (3) by definition, do not satisfy the triangle inequality, and hence are semi-metrics, then Equations (6) and (7) are also semi-metrics between $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$.

We note that it is easy to satisfy all the metric properties for Equations (4) and (5) by taking their square root, and ensuring that $\mathbf{\Psi}$ and $\mathbf{\Gamma}^{-1}$ are PD. In practice, the positive definiteness of $\mathbf{\Psi}$ and $\mathbf{\Gamma}^{-1}$ can be achieved by ensuring that $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are PD. For high dimensional data, shrinkage estimators for covariance matrices [5] are usually used to estimate regularized versions of $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$. These estimates are statistically efficient, PD, and well conditioned[5].

[5] See for instance [5] and its affiliated references for a nice overview on these methods, and some recent developments in this direction.

The problem, however, remains with Equations (6) and (7). Covariance matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$ are elements of $\mathbf{S}_{++}^{p\times p}$, which is a metric space with a defined metric for its elements. The semi-metrics in Equations (6) and (7), although naturally derived from divergence measures [3, 6], do not define proper metrics for $\mathbf{S}_{++}^{p\times p}$, and hence violate its geometric properties.

The set of symmetric PD matrices is a set of geometric objects that define the Riemannian manifold $\mathbb{S}_{++}^{p\times p}$. A Riemannian manifold is a differentiable manifold equipped with an inner product that induces a natural distance metric, or a Riemannian metric between all its elements. Förstner and Moonen [7] and independently X. Pennec [17] derived this metric for $\mathbb{S}_{++}^{p\times p}$, however its history goes back to C. R. Rao in 1945 [18]. For $\mathbf{\Sigma}_1, \mathbf{\Sigma}_2 \in \mathbb{S}_{++}^{p\times p}$, the Riemannian metric is defined as:

$$d_{\mathcal{R}}(\mathbf{\Sigma}_1, \mathbf{\Sigma}_2) = \left(\textstyle\sum_{j=1}^{p} \log^2 \lambda_j\right)^{\frac{1}{2}}, \tag{8}$$

where $\operatorname{diag}(\lambda_1, \ldots, \lambda_p) = \mathbf{\Lambda}$ is the generalized eigenvalue matrix for the generalized eigenvalue problem (GEP): $\mathbf{\Sigma}_1\mathbf{V} = \mathbf{\Lambda}\mathbf{\Sigma}_2\mathbf{V}$, and $\mathbf{V}$ is the column matrix of its generalized eigenvectors. Note that $d_{\mathcal{R}}$ satisfies all metric axioms and is invariant to inversion and to affine transformations of the coordinate system [7].

2.1 Modifying $d_J(\mathcal{G}_1, \mathcal{G}_2)$ and $d_B(\mathcal{G}_1, \mathcal{G}_2)$

Modifying the divergence measures $d_J(\mathcal{G}_i, \mathcal{G}_j)$ and $d_B(\mathcal{G}_i, \mathcal{G}_j)$ in Equations (1) and (3) respectively, will rely on (i) their special structure which decomposes the difference between two Gaussian densities into the difference between their first and second order moments, and (ii) the fact that the second term in Equations (1) and (3) is independent from the means $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$. This split of the Gaussian parameters encourages us to exchange the second term in $d_J(\mathcal{G}_1, \mathcal{G}_2)$ and $d_B(\mathcal{G}_1, \mathcal{G}_2)$, i.e. the semi-metrics for covariance matrices in Equations (6) and (7), with the Riemannian metric $d_{\mathcal{R}}$ in Equation (8). More specifically, we propose the following metrics as measures for the difference between two Gaussians:

$$d_{J\mathcal{R}}(\mathcal{G}_1, \mathcal{G}_2) = (\mathbf{u}^\top \boldsymbol{\varPsi} \mathbf{u})^{\frac{1}{2}} + d_{\mathcal{R}}(\boldsymbol{\varSigma}_1, \boldsymbol{\varSigma}_2), \quad \text{and} \tag{9}$$

$$d_{B\mathcal{R}}(\mathcal{G}_1, \mathcal{G}_2) = (\mathbf{u}^\top \boldsymbol{\varGamma}^{-1} \mathbf{u})^{\frac{1}{2}} + d_{\mathcal{R}}(\boldsymbol{\varSigma}_1, \boldsymbol{\varSigma}_2), \tag{10}$$

where $\boldsymbol{\varPsi} \succ 0$, and $\boldsymbol{\varGamma}^{-1} \succ 0$. Note that each term of the proposed measures satisfy all metric axioms. Further, Equations (9) and (10) keep the same structure and characteristics of Equations (1) and (3); in particular the second term is independent from $\boldsymbol{\mu}_1$ and $\boldsymbol{\mu}_2$. If $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \boldsymbol{\mu}$ then Equations (9) and (10) reduce to the Riemannian metric $d_{\mathcal{R}}$ in Equation (8). If $\mathbf{\Sigma}_1 = \mathbf{\Sigma}_2 = \mathbf{\Sigma}$, then Equations (9) and (10) will yield the exact GQD with symmetric PD matrices $\mathbf{\Psi}$ and $\mathbf{\Gamma}^{-1}$ respectively, and if $\mathbf{\Sigma} = \mathbf{I}$, then the two metrics will yield the Euclidean distance. In the case when $\boldsymbol{\mu}_1 \neq \boldsymbol{\mu}_2$ and $\mathbf{\Sigma}_1 \neq \mathbf{\Sigma}_2$, an α–weighted version of (9) and (10) can be expressed as:

$$d_{J\mathcal{R}}(\mathcal{G}_1, \mathcal{G}_2; \alpha) = \alpha(\mathbf{u}^\top \boldsymbol{\varPsi} \mathbf{u})^{\frac{1}{2}} + (1-\alpha) d_{\mathcal{R}}(\boldsymbol{\varSigma}_1, \boldsymbol{\varSigma}_2), \quad \text{and}$$

$$d_{B\mathcal{R}}(\mathcal{G}_1, \mathcal{G}_2; \alpha) = \alpha(\mathbf{u}^\top \boldsymbol{\varGamma}^{-1} \mathbf{u})^{\frac{1}{2}} + (1-\alpha) d_{\mathcal{R}}(\boldsymbol{\varSigma}_1, \boldsymbol{\varSigma}_2),$$

where $\alpha \in (0,1)$ weights the contribution (or importance) of each term in $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$. Note that when the α–weighted version of the measures are plugged in a learning algorithm, α can be optimized by methods of cross validation, or jointly optimized with the intensity/shrinkage parameters used to regularize the covariance matrices $\mathbf{\Sigma}_1$ and $\mathbf{\Sigma}_2$.

3 Manifold Learning with Divergence Measures

Given a set vectors $\mathcal{X} = \{\mathbf{x}_i\}_{i=1}^n$, $\mathbf{x}_i \in \mathbb{R}^p$, manifold learning algorithms [20, 4] construct a neighbourhood graph in which the input points $\mathbf{x}_i$ act as its vertices. This graph is an estimate for the topology of an underlying low dimensional manifold on which the data are assumed to lie on. The learning algorithm then, tries to unfold this manifold – while preserving some local information – to partition the graph (as in clustering), or to redefine metric information (as in dimensionality reduction). The algorithm's output is the set $\mathcal{Y} = \{\mathbf{y}_i\}_{i=1}^n$ that lives in a subspace of dimensionality $p_0 \ll p$, where $\mathbf{y}_i \in \mathbb{R}^{p_0}$ is the embedding of the input $\mathbf{x}_i$.

A different setting occurs when each vertex v_i on the graph represents a set $\mathcal{S}_i$, where $\mathcal{S}_i = \{\mathbf{x}_j^i\}_{j=1}^{n_j}$ is a set of vectors. For instance, $\mathcal{S}_i$ can be the feature vectors describing a multimedia file [16], an image [10], or a short video clip [1]. In these settings, each $\mathcal{S}_i$ is modelled as a Gaussian distribution $\mathcal{G}_i$, and the pairwise dissimilarity between all the Gaussians $\{\mathcal{G}_i\}_{i=1}^n$ is measured using divergence measures. This, however, turns the problem into obtaining a low dimensional embedding for the family of Gaussians $\{\mathcal{G}_i\}_{i=1}^n$. Again, the algorithm's output is the set $\mathcal{Y} = \{\mathbf{y}_i\}_{i=1}^n$, with $\mathbf{y}_i \in \mathbb{R}^{p_0}$ being the low dimensional embedding (representation) of the Gaussian $\mathcal{G}_i$.

Before proceeding to obtain such an embedding, it is important to understand how the metric properties of divergence measures can affect the graph embedding process of these algorithms. To illustrate these properties, we pick two different types of algorithms: cMDS [21] and LEM [4].

It turns out that the metric properties of divergence measures are intimately related to the positive semi-definiteness of the affinity matrix $\mathbf{A} \in \mathbb{R}^{n \times n}$ extracted from the graph's adjacency matrix. Let $\mathbf{D} \in \mathbb{R}^{n \times n}$ be the matrix of pairwise divergences where $\mathbf{D}_{ij} = div(\mathcal{G}_i, \mathcal{G}_j)$, $\forall i, j$, and div is a symmetric divergence measure.

For cMDS, the affinity matrix $\mathbf{A}$ is defined as $\mathbf{A}_{ij} = -\frac{1}{2}\mathbf{D}_{ij}^2$, $\forall i, j$. The matrix $\mathbf{A}$ is guaranteed to be PSD *if and only if* $div(\mathcal{G}_i, \mathcal{G}_j)$ is a metric; in particular satisfies the triangle inequality. This result is due to Theorem (3) in [21] and Theorem (4) in [8]. Therefore, div in the case of cMDS can be d_H, d_{JS}, $d_{J\mathcal{R}}$, or $d_{B\mathcal{R}}$ since they are all metrics.

For LEM, and for input vectors $\mathbf{x}_i, \mathbf{x}_j$, the affinity matrix $\mathbf{A}$ is defined as $\mathbf{A}_{ij} = K(\mathbf{x}_i, \mathbf{x}_j)$, $\forall i, j$, where K is a symmetric PSD kernel that measures the similarity between $\mathbf{x}_i$ and $\mathbf{x}_j$. From Mercer kernels [15], it is known that $\mathbf{A}$ is PSD *if and only if* K is symmetric and PSD. Note that for any two probability distributions P_1 and P_2, and by definition of divergence [3], $div(P_1, P_2) \geq 0$, and equality only holds when $P_1 = P_2$. Hence $div(P_1, P_2)$ is PSD by definition and it can also be symmetric as d_J, d_B, d_H, d_{JS}, $d_{J\mathcal{R}}$, and $d_{B\mathcal{R}}$.

A possible kernel for $\mathcal{G}_i$ and $\mathcal{G}_j$ using a symmetric div is: $K(\mathcal{G}_i, \mathcal{G}_j) = \exp\{-\frac{1}{\sigma} div(\mathcal{G}_i, \mathcal{G}_j)\} = \exp\{-\frac{1}{\sigma}\mathbf{D}_{ij}\}$, where $\sigma > 0$ is a parameter that scales the affinity between two densities. Since div is PSD and symmetric, then $K(\mathcal{G}_i, \mathcal{G}_j)$ is PSD and symmetric as well. This simple fact is due to Theorems (2) and (4) in [19], and a discussion on these particular kernels can be found in [2]. Further, if div is a metric, then the isometric embedding $\exp\{-div\}$ will result in a metric space (see footnote in pp. 525 of [19]), and the resulting embedding of LEM will be isometric as well. Therefore, for LEM, a symmetric PSD affinity matrix can be defined as $\mathbf{A}_{ij} = K(\mathcal{G}_i, \mathcal{G}_j)$, $\forall i, j$, and using any symmetric div to define the kernel K. Note that LEM is more flexible than cMDS since it only requires a symmetric divergence, while cMDS needs all metric axioms to be satisfied.

4 Experiments

To test the validity and efficacy of the proposed measures $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$, and to compare their performance to d_J, d_B, and d_H, we conduct a set of experiments in the context of clustering human motion from video sequences. Our main objective from these experiments is to show that the proposed measures $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$ can consistently outperform other divergence measures in a nontrivial and rather challenging task such as human motion clustering in video data.

For the purpose of our experiments, we use the KTH data set for human action recognition[6]. The data set consists of video clips for 6 types of human actions (boxing, hand clapping, hand waving, jogging, running, and walking) performed by 25 subjects in 4 different scenarios (outdoors, outdoors with scale variation, outdoor with different clothes, and indoors), resulting in a total number of video clips $n = 6 \times 25 \times 4 = 600$. All sequences were taken over homogeneous backgrounds with a static camera with a frame rate of 25 fps. The spatial resolution of the videos is 160×120, and each clip has a length of 20 seconds on average.

4.1 Representing Motion as Sets of Vectors

In these experiments, a long video sequence $V = \{\mathbf{F}_t\}_{t=1}^{\tau}$ with intensity frames $\mathbf{F}_t$ is divided into very short video clips $VClip$ of equal length k where it is assumed that an apparent smallest human action can occur; i.e. $V = \{\,VClip_i\}_{i=1}^{n}$. Depending on the video sampling rate, $k = \{20, 25, 30, 35\}$ frames/clip. This is the first column in Tables in (1) and (2).

To extract the motion information, a dense optical flow is computed for each video clip using the Lucas-Kanade algorithm [13][7], resulting in a large set of spatio-temporal gradients vectors describing the motion of pixels in each frame. To capture the motion information encoded in the gradient direction,

[6] http://www.nada.kth.se/cvap/actions/

[7] Implemented in Piotr's Image and Video Toolbox for Matlab http://vision.ucsd.edu/ pdollar/toolbox/doc/

first we apply an adaptive threshold based on the norm of the gradient vectors to eliminate all vectors resulting from slight illumination changes and camera jitter. Second, each video frame is divided into $h \times w$ blocks – typically 3×3 and 4×4 – and the motion in each block is encoded by an m–bins histogram of gradient orientations. In all our experiments, m is set to 4 and 8 bins. The histograms of all blocks for one frame are concatenated to form one vector of dimensionality $p = m \times h \times w$. Therefore, a video clip $VClip_i$ with k frames is finally represented as a set $\mathcal{S}_i = \{\mathbf{x}_1^i, \ldots, \mathbf{x}_k^i\}$, where $\mathbf{x}_j^i$ is a p-dimensional vector of the concatenated histograms of frame j. Last, for each subject, the video clips for the 6 actions from one scenario were concatenated to form one long video sequence. This resulted in $25 \times 4 = 100$ long video sequences that were used in our experiments. To validate the accuracy of clustering, each video frame was labeled with the type of action it contains.

4.2 Experimental Setting

Once the motion information in video V is represented as a family of sets $\{\mathcal{S}_i\}_{i=1}^n$, motion clustering tries to group together video clips (or sets) with similar motion vectors. To this end, we use a recently proposed framework for learning over sets of vectors [1] to obtain such a clustering for the $\mathcal{S}_i$'s. In this framework, each $\mathcal{S}_i = \{\mathbf{x}_j^i\}_{j=1}^{n_i}$ is modelled as a Gaussian distribution $\mathcal{G}_i$ with mean vector $\hat{\boldsymbol{\mu}}_i = \frac{1}{n_i}\sum_{j=1}^{n_i} \mathbf{x}_j^i$, and a covariance matrix $\hat{\boldsymbol{\Sigma}}_i = \frac{1}{n_i-1}\sum_{j=1}^{n_i}(\mathbf{x}_j^i - \hat{\boldsymbol{\mu}}_i)(\mathbf{x}_j^i - \hat{\boldsymbol{\mu}}_i)^\top + \gamma\mathbf{I}$, where γ is a regularization parameter. This forms the family of Gaussians $\{\mathcal{G}_i\}_{i=1}^n$ which represents the motion in V. Note that regularization is necessary for high dimensional data especially when $n_i \leq p$ (rank deficient covariance matrix) to avoid over fitting, leverage noise effect in the data, and outlier reliance[8].

Using cMDS and LEM together with the divergence measures discussed here, d_J, d_B, d_H, $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$, we obtain a low dimensional embedding for the family of Gaussians as the set $\{\mathbf{y}_i\}_{i=1}^n$, where $\mathbf{y}_i \in \mathbb{R}^{p_0}$, and $p_0 \ll p$. Finally, the k-Means clustering is run on the data set $\{\mathbf{y}_i\}_{i=1}^n$. To summarize, a video sequence goes through the following transformations:
$V \longmapsto \{VClip_i\}_{i=1}^n \longmapsto \{\mathcal{S}_i\}_{i=1}^n \longmapsto \{\mathcal{G}_i\}_{i=1}^n \longmapsto \{\mathbf{y}_i\}_{i=1}^n$.

The dimensionality p_0 of the embedding space is a hyperparameter for cMDS and LEM. For cMDS this is allowed to change from 2 up to 100 dimensions, while for LEM it is usually set equal to the number of clusters which is 6 in this case [14]. This is due to our *a priori* knowledge that there are 6 types of motion in each video. Another hyperparameter to optimize for LEM is the kernel width σ which was allowed to take 4 different values from all the pairwise divergences; the median, 0.25, 0.75, and 0.9 of the quantile.

For the k-Means algorithm, the number of clusters k was set to 6, and to avoid local minima, the algorithm was run with 30 different initializations and the run with the minimum sum of squared distances was selected as the final result for clustering. The clustering accuracy here is measured using the Hungarian score used in [22] which finds the maximum matching between the true labeling

[8] In all our experiments $\gamma = 1$.

Table 1: Average clustering accuracy (with standard deviations) over 100 video sequences in 4 different embedding spaces obtained using cMDS+d_J, cMDS+d_B, cMDS+d_H, and cMDS+$d_{J\mathcal{R}}$.

	$p = m \times h \times w = 8 \times 3 \times 3$			
frames/clip	cMDS+d_J	cMDS+d_B	cMDS+d_H	cMDS+$d_{J\mathcal{R}}$
20	70.9 (11.9)	71.0 (12.0)	75.5 (12.1)	**80.3 (10.9)**
25	62.8 (10.9)	62.8 (11.0)	68.2 (12.3)	**75.5 (13.1)**
30	66.7 (11.7)	66.7 (11.8)	71.5 (12.7)	**77.4 (12.7)**
35	62.8 (10.9)	62.8 (11.1)	68.2 (12.3)	**75.3 (13.1)**
	$p = m \times h \times w = 8 \times 4 \times 4$			
frames/clip	cMDS+d_J	cMDS+d_B	cMDS+d_H	cMDS+$d_{J\mathcal{R}}$
20	68.3 (12.1)	68.9 (11.6)	74.2 (12.0)	**79.5 (11.7)**
25	66.5 (12.0)	66.5 (12.4)	72.5 (12.2)	**78.6 (12.1)**
30	61.9 (10.9)	63.0 (10.6)	68.9 (11.4)	**75.5 (12.3)**
35	71.3 (12.1)	71.8 (12.3)	76.5 (11.8)	**80.7 (10.1)**

of each video clip and the labeling produced by the clustering algorithm. Note that this is the accuracy for clustering one and only one long video sequence. The values recorded in Columns 2, 3, 4, and 5 in Tables (1) and (2) are the average accuracies (with standard deviations) over the 100 video sequences created for these experiments (§4.2). During these experiments, it was noted that the performance for $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$ are very similar under both algorithms, and hence, due to space limitations, we show the results of cMDS+$d_{J\mathcal{R}}$ in Table (1) and the results for LEM+$d_{B\mathcal{R}}$ in Table (2).

4.3 Analysis of the Results

Our hypothesis, before running the experiments, is that clustering accuracy in the embedding space obtained through the modified divergences $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$ will be higher than the clustering accuracy in the embedding spaces obtained by other divergence measures. Note that the k-Means accuracy here is a quantitative indicator on the quality of the embedding and its capability to define clusters, or regions of high density (manifolds), which correspond to clusters of different motion types. Therefore, each embedding space is optimized to maximize the clustering accuracy, and then the highest accuracy obtained is compared against all other highest accuracies of other embedding spaces.

Tables (1) and (2) show that, under the embeddings of cMDS and LEM with $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$, the clustering accuracy is consistently superior to the accuracy of both algorithms with other divergence measures. This implies that the embedding spaces obtained via the new proposed measures can better characterize the cluster structure in the data, and hence the high clustering accuracies in Tables (1) and (2). Another observation to note from Tables (1) and (2) is that the clustering accuracies under the embedding of cMDS and LEM with d_H (which is a metric) are higher than the accuracies obtained with the same algorithms

Table 2: Average clustering accuracy (with standard deviations) over 100 video sequences in 4 different embedding spaces obtained using LEM+d_J, LEM+d_B, LEM+d_H, and LEM+$d_{B\mathcal{R}}$.

	$p = m \times h \times w = 8 \times 3 \times 3$			
frames/clip	LEM+d_J	LEM+d_B	LEM+d_H	LEM+$d_{B\mathcal{R}}$
20	55.7 (11.2)	56.0 (10.9)	60.1 (11.5)	**65.1 (13.2)**
25	58.2 (12.0)	58.1 (11.9)	63.6 (13.1)	**69.6 (13.6)**
30	60.0 (12.7)	59.9 (12.6)	64.8 (12.9)	**70.3 (13.4)**
35	63.0 (13.3)	62.9 (13.3)	67.4 (13.1)	**71.8 (13.6)**
	$p = m \times h \times w = 8 \times 4 \times 4$			
frames/clip	LEM+d_J	LEM+d_B	LEM+d_H	LEM+$d_{B\mathcal{R}}$
20	54.0 (12.5)	54.6 (12.7)	60.8 (12.2)	**66.3 (12.7)**
25	57.7 (14.0)	57.7 (13.9)	64.7 (13.2)	**69.5 (13.2)**
30	59.5 (13.4)	59.5 (13.2)	66.3 (12.5)	**70.5 (12.6)**
35	59.5 (13.4)	59.5 (13.2)	66.3 (12.5)	**70.5 (12.6)**

but using d_J and d_B. Again, this implies that the obtained embedding space via d_H can better characterize the cluster structure in the data. However, when comparing d_H on one hand, versus $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$ on the other, we note that the embeddings obtained via d_H yield consistently lower performance than $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$ do. In our understanding, this is due to its measure for the difference between covariance matrices[9], $(2 - 2|\boldsymbol{\Gamma}|^{-\frac{1}{2}}|\boldsymbol{\Sigma}_1|^{\frac{1}{4}}|\boldsymbol{\Sigma}_2|^{\frac{1}{4}})^{\frac{1}{2}}$, which is not a metric on $\mathbb{S}_{++}^{p\times p}$ and hence it violates its geometry.

The low performance for d_J and d_B with both algorithms when compared to the other divergence measures is again due to their lack of metric properties (in particular the triangle inequality), which in turn impacts the characteristics preserved (or relinquished) by the embedding procedure. Note that the difference in performance is more clear for the cMDS case in Table (1). None of d_J and d_B them is a true metric, and hence, they can result in embeddings that do not preserve the relative dissimilarities among all objects assigned to the graph's vertices. This can easily collapse a group of objects to be very close to each other in the embedding space thereby misleading the k–Means clustering algorithm.

In summary, it can be seen that, on the same data sets, and despite the differences between cMDS and LEM as dimensionality reduction algorithms, both algorithms showed consistent and identical behaviour in terms of relative responses to the different divergence measures discussed here which validates our hypothesis with regards to the proposed metrics $d_{J\mathcal{R}}$ and $d_{B\mathcal{R}}$.

5 Concluding Remarks

Our research presented here is motivated by the following question: Do metric properties of divergence measures have an impact on the output hypothesis of

[9] by setting $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \boldsymbol{\mu}$ in $d_H(\mathcal{G}_1, \mathcal{G}_2)$.

a learning algorithm, and hence on its performance? In this paper, we tried to answer this question through the following: First, we analyzed some well known divergence measures for the particular case of multivariate Gaussian densities since they are pervasive in machine learning and pattern recognition. Second, based on our analysis, we proposed a simple modification to two well known divergence measures for Gaussian densities. The modification led to two new distance metrics between Gaussian densities in which their constituting elements respect the geometry of their corresponding spaces. Next, we showed how the metric properties can impact the graph embedding process of manifold learning algorithms, and demonstrated empirically how the proposed new metrics yield better embedding spaces in a totally unsupervised manner.

Our study suggests that metric properties of divergence measures constitute an important aspect of the model selection question for divergence based learning algorithms. Further, the proposed metrics developed here are not restricted to manifold learning algorithms, and they can be used in various contexts, such as metric learning, discriminant analysis, and feature selection to mention a few.

References

1. Abou-Moustafa, K., Ferrie, F.: A framework for hypothesis learning over sets of vectors. In: Proc. of 9th SIGKDD Workshop on Mining and Learning with Graphs. pp. 335–344. ACM (2011)
2. Abou-Moustafa, K., Shah, M., Torre, F.D.L., Ferrie, F.: Relaxed exponenrial kernels for unsupervised learning. In: LNCS 6835, Pattern Recognition, Proc. of the 33rd DAGM Symp. pp. 335–344. Springer (2011)
3. Ali, S.M., Silvey, S.D.: A general class of coefficients of divergence of one distribution from another. J. of the Royal Statistical Society. *Series B* 28(1), 131–142 (1966)
4. Belkin, M., Niyogi, P.: Laplacian eigenmaps and spectral techniques for data representation. Neural Computation 15, 1373–1396 (2003)
5. Cao, G., Bachega, L., Bouman, C.: The sparse matrix transform for covariance estimation and analysis of high dimensional signals. IEEE. Trans. on Image Processing 20(3), 625 – 640 (Mar 2011)
6. Csiszár, I.: Information–type measures of difference of probability distributions and indirect observations. Studia Scientiarium Mathematicarum Hungarica 2, 299–318 (1967)
7. Förstner, W., Moonen, B.: A metric for covariance matrices. Tech. rep., Dept. of Geodesy and Geo–Informatics, Stuttgart University (1999)
8. Gower, J., Legendre, P.: Metric and Euclidean properties of dissimilarity coefficients. J. of Classification 3, 5–48 (1986)
9. Kailath, T.: The divergence and Bhattacharyya distance measures in signal selection. IEEE Trans. on Communication Technology 15(1), 52–60 (1967)
10. Kondor, R., Jebara, T.: A kernel between sets of vectors. In: ACM Proc. of ICML (2003)
11. Kreyszig, E. (ed.): Introductory functional Analysis with Applications. Wiley Classics Library (1989)
12. Kullback, S.: Information Theory and Statistics – Dover Edition. Dover, New York (1997)

13. Lucas, B., Kanade, T.: An iterative image registration technique with an application to stereo vision. In: Proc. of IJCAI. pp. 674–679 (1981)
14. Luxburg, U.v.: A tutorial on spectral clustering. Statistics and Computing 17(4), 395–416 (2007)
15. Mercer, J.: Functions of positive and negative type, and their connection with the theory of integral equations. Philosophical Trans. of the Royal Society of London. Series A 209, 415–446 (1909)
16. Moreno, P., Ho, P., Vasconcelos, N.: A Kullback–Leibler divergence based kernel for svm classification in multimedia applications. In: NIPS 16 (2003)
17. Pennec, X., Fillard, P., Ayache, N.: A Riemannian Framework for Tensor Computing. Tech. Rep. RR-5255, INRIA (7 2004)
18. Rao, C.R.: Information and the accuracy attainable in the estimation of statistical parameters. Bull. Calcutta Math. Soc. (58), 326–337 (1945)
19. Schoenberg, I.: Metric spaces and positive definite functions. Trans. of the American Mathematical Society 44(3), 522–536 (1938)
20. Tenenbaum, J., de Silva, V., Langford, J.: A global geometric framework for nonlinear dimensionality reduction. Science 290(5500), 2319–2323 (November 2000)
21. Young, G., Householder, A.: Discussion of a set of points in terms of their mutual distances. Psychometrika 3(1), 19–22 (1938)
22. Zha, H., Ding, C., Gu, M., He, X., Simon, H.: Spectral relaxation for k–means clustering. In: NIPS 13. MIT Press (2001)