Signal Processing 182 (2021) 107942

Contents lists available at ScienceDirect

Signal Processing

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

L1-norm unsupervised Fukunaga-Koontz transform☆

José Luis Camargo[a], Rubén Martín-Clemente[a,*], Susana Hornillo-Mellado[a], Vicente Zarzoso[b]

[a] Signal Theory and Communications Department, Universidad de Sevilla, Spain
[b] Université Côte d'Azur, CNRS, I3S Laboratory, Sophia Antipolis, France

ARTICLE INFO

Article history:
Received 2 May 2020
Revised 21 October 2020
Accepted 11 December 2020
Available online 24 December 2020

MSC:
02.50.Sk
43.60.Cg
07.50.Qx

Keywords:
Fukunaga-Koontz
Common spatial patterns
Tuned-based functions
L1-PCA

ABSTRACT

The Fukunaga-Koontz transform (FKT) is a powerful supervised feature extraction method used in two-class recognition problems, particularly when the classes have equal mean vectors but different covariance matrices. The present work proves that it is also possible to perform the FKT in an unsupervised manner, sparing the need for labeled data, by using a variant of L1-norm Principal Component Analysis (L1-PCA) that minimizes the L1-norm in the feature space. Rigorous proof is given in the case of data drawn from a mixture of Gaussians. A working iterative algorithm based on gradient-descent in the Stiefel manifold is put forward to perform L1-norm minimization with orthogonal constraints. A number of numerical experiments on synthetic and real data confirm the theoretical findings and the good convergence characteristics of the proposed algorithm.

© 2021 The Authors. Published by Elsevier B.V.
This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

1. Introduction

The Fukunaga-Koontz transform (FKT) is a popular feature-extraction method used in binary classification problems [1]. It projects the data onto directions along which the variance is much larger for one class than for the other. The classification rule thus exploits the difference in variance between the two projected classes.

The true potential of the FKT is revealed when the two classes share the same mean vector, giving overlapping sets of data [2,3]. For equal-mean class distributions, reference [4] shows that the FKT is equivalent to the optimal Chernoff criterion introduced in [5], and thus preserves after the projection as much as possible of the Chernoff distance between both populations [6]. The FKT is also closely related to optimal Linear and Quadratic Discriminant Analysis and the Generalized Singular Value Decomposition, as has been shown in the literature [2,7,8]. Thanks to all its properties, the FKT has been successfully applied to image classification problems (where it is also known as the method of the 'tuned based functions') [9–13] and in EEG signal processing under the name of 'common spatial patterns' [14,15], as well as in other areas of practical interest [16,17].

Our contribution is to show, for the first time, a link between the FKT and a variant of Principal Component Analysis (PCA) called L1-PCA, which is receiving increasing interest due to its resistance to outliers [18–20] and its connections to Independent Component Analysis and Linear Discriminant Analysis [21,22]. L1-PCA linearly projects the data onto a few dimensions that maximize the absolute value of the projected data points. Just changing the word 'maximize' to 'minimize', while retaining the absolute value as objective function, this paper shows that it is also possible to calculate the Fukunaga-Koontz directions of projection. A rigorous proof of this result is given for the case of Gaussian populations with zero mean but different covariance matrices, whereas for non-Gaussian data we provide an experimental demonstration of this result. We only require the raw data points be pre-whitened to remove their covariance structure. The theoretical importance of the above result is that it relates these two apparently disparate techniques, allowing us to re-interpret the absolute value as a feature-extraction criterion in binary classification problems, which opens new lines of research in this area.

Furthermore, apart from its theoretical interest, this result also has practical relevance because the standard FKT is a supervised

☆ This work is funded by the research project ACACIA (refno. US-1264994 US/JUNTA/FEDER, UE) awarded by Fondo Europeo de Desarrollo Regional (FEDER) and Junta de Andalucía (Consejería de Economía, Conocimiento, Empresas y Universidad).

* Corresponding author.
E-mail addresses: jlcamargo@yahoo.es (J.L. Camargo), ruben@us.es (R. Martín-Clemente), susanah@us.es (S. Hornillo-Mellado), vicente.zarzoso@univ-cotedazur.fr (V. Zarzoso).

https://doi.org/10.1016/j.sigpro.2020.107942
0165-1684/© 2021 The Authors. Published by Elsevier B.V. This is an open access article under the CC BY-NC-ND license (http://creativecommons.org/licenses/by-nc-nd/4.0/)

Table 1
Notation and symbols used in this paper.

$\mathbf{0}$	vector of zeros
$\boldsymbol{I}$	identity matrix
$(\cdot)^\dagger$	matrix transpose operator
$\det(\cdot)$, $\mathrm{trace}(\cdot)$	determinant and trace of a matrix
X	p-dimensional random variable
$\boldsymbol{x}$	observation of X
$P(\mathcal{C}_i)$	probability of the observed data being drawn from class $\mathcal{C}_i$
$\mathrm{E}\{\cdot\}$	mathematical expectation
$\mathrm{E}\{\cdot \mid \mathcal{C}_i\}$	conditional expectation given the class $\mathcal{C}_i$
$f(\cdot \mid \mathcal{C}_i)$	conditional probability density function (pdf) given $\mathcal{C}_i$
$\boldsymbol{\mu}_i = \mathrm{E}\{X \mid \mathcal{C}_i\}$	mean of class i
$\boldsymbol{\Sigma}_i$	covariance of class i
$\|\boldsymbol{x}\|$	L2-norm of vector $\boldsymbol{x}$
$\|\boldsymbol{X}\|$	Frobenius norm of matrix $\boldsymbol{X}$

technique, which requires a training set of correctly class-labeled data points to estimate the parameters of the transformation; however, by contrast, minimizing the absolute value can be performed in a fully *unsupervised* fashion, making unnecessary the acquisition of training data and opening the door to the computation of the FKT in the same way.

The paper is organized as follows: we first present in Section 2 some general assumptions made in the paper. Section 3 briefly reviews the FKT. In Sections 4 and 5, we state our main results and propose a numerical algorithm for unsupervised FKT based on L1-norm minimization. Section 6 provides a number of numerical experiments that validate our findings in a variety of scenarios. Finally, Section 7 brings the paper to an end. Note that mathematical proofs have been deferred to the Appendices for the reader's convenience.

2. Preliminaries: Notation and basic hypotheses

The following assumptions hold throughout the paper. Let $X \in \mathbb{R}^p$ be a random vector whose samples $\boldsymbol{x}$ are drawn at random from one of two populations, $\mathcal{C}_1$ and $\mathcal{C}_2$. We suppose that $\mathcal{C}_1$ and $\mathcal{C}_2$ have common mean vectors $\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2$ but different covariance matrices $\boldsymbol{\Sigma}_1 \neq \boldsymbol{\Sigma}_2$. For simplicity, we also suppose that these matrices do not have repeated eigenvalues. Other symbols and notations used in this paper can be found in Table 1.

It is assumed as well that the data have been *centered* (by subtracting the mean across all observations) and *whitened* (or *sphered*). *Centering* implies that

$$\boldsymbol{\mu}_1 = \boldsymbol{\mu}_2 = \mathbf{0}. \tag{1}$$

Whitening consists in transforming the data to have identity covariance matrix:

$$\begin{aligned}\boldsymbol{\Sigma} &= \mathrm{E}\{X X^\dagger\} \\ &= P(\mathcal{C}_1)\mathrm{E}\{X X^\dagger \mid \mathcal{C}_1\} + P(\mathcal{C}_2)\mathrm{E}\{X X^\dagger \mid \mathcal{C}_2\} \\ &= P(\mathcal{C}_1)\boldsymbol{\Sigma}_1 + P(\mathcal{C}_2)\boldsymbol{\Sigma}_2 \\ &= \boldsymbol{I}. \end{aligned} \tag{2}$$

Like centering, whitening can be assumed without any loss of generality: it can be always fulfilled by a simple pre-processing step (see Section 5.1). Whitening is useful, as we will show in the next Section, because it intertwines the class covariances as follows:

Property 1. *After whitening,*

$$\boldsymbol{\Sigma}_1 = \frac{1}{P(\mathcal{C}_1)}\,[\boldsymbol{I} - P(\mathcal{C}_2)\boldsymbol{\Sigma}_2]. \tag{3}$$

Eq. (3) readily follows from Eq. (2). Thanks to this intertwining, we will see in the next Section that $\mathcal{C}_1$ and $\mathcal{C}_2$ lie (approximately) in orthogonal subspaces.

3. The Fukunaga-Koontz transform

Let $(\lambda, \boldsymbol{v})$ be any eigenpair of $\boldsymbol{\Sigma}_1$, i.e.,

$$\boldsymbol{\Sigma}_1 \boldsymbol{v} = \lambda\, \boldsymbol{v}. \tag{4}$$

Using (3), it readily follows that

$$\frac{1}{P(\mathcal{C}_1)}\,[\boldsymbol{I} - P(\mathcal{C}_2)\boldsymbol{\Sigma}_2]\boldsymbol{v} = \lambda\, \boldsymbol{v}$$

and hence

$$\boldsymbol{\Sigma}_2 \boldsymbol{v} = \frac{1 - P(\mathcal{C}_1)\,\lambda}{P(\mathcal{C}_2)}\,\boldsymbol{v}.$$

That is: if $\boldsymbol{v}$ is any eigenvector of $\boldsymbol{\Sigma}_1$ with eigenvalue λ, then $\boldsymbol{v}$ is also an eigenvector of $\boldsymbol{\Sigma}_2$ with eigenvalue

$$\mu = \frac{1 - P(\mathcal{C}_1)\,\lambda}{P(\mathcal{C}_2)}.$$

This transformation is strictly decreasing: if the eigenvalues λ_i of $\boldsymbol{\Sigma}_1$ are ordered from largest to smallest as

$$\lambda_1 > \lambda_2 > \ldots > \lambda_p,$$

it follows that the corresponding eigenvalues of $\boldsymbol{\Sigma}_2$ are reversely ordered as

$$\mu_1 < \mu_2 < \ldots < \mu_p$$

so that the dominant eigenvectors of $\boldsymbol{\Sigma}_1$ are the least dominant eigenvectors of $\boldsymbol{\Sigma}_2$ and *vice versa*. In the language of classical PCA [23], the directions in which the data from class 1 vary the most are also the directions where class 2 varies the least. The opposite is also true: the directions of greatest variance for class 2 are those of lowest variance for class 1. This makes the two classes easier to distinguish.

Furthermore, the averaged squared distance between the data points from one class and the subspace spanned by the dominant eigenvectors of their class covariance matrix is minimal. This is, in this sense, the best-fitting subspace [23].

Feature extraction and classification can be based on exploiting all these properties. The FKT transforms each data point by orthogonally projecting it onto the span of the eigenvectors of $\boldsymbol{\Sigma}_1$ corresponding to the largest and smallest eigenvalues. If the data point is closer to the subspace spanned by the first few dominant eigenvectors than to the subspace spanned by the least dominant eigenvectors, we can assume the presence of a sample of $\mathcal{C}_1$. The opposite suggests allocating it to class $\mathcal{C}_2$. Several variants of this basic approach have been also proposed, see [9–13].

Note finally that, in standard FKT, matrices $\boldsymbol{\Sigma}_1$ or $\boldsymbol{\Sigma}_2$ have to be estimated *a priori* from a set labelled samples. We remark that, for this reason, the standard FKT is a *supervised* technique.

4. Main contribution: Unsupervised FKT via L1-norm minimization

Unsupervised calculation of the FKT is however possible by minimizing the L1-norm of the projection: in this Section we prove this property in the Gaussian case. Gaussian models are justified by their simplicity and ability to produce accurate results in practice, even when violated. In particular, we make the usual assumption that $f(\boldsymbol{x}|\mathcal{C}_i)$ is a p-variate normal density function of the form

$$f(\boldsymbol{x}|\mathcal{C}_i) = (2\pi)^{-\frac{p}{2}} \det(\boldsymbol{\Sigma}_i)^{-\frac{1}{2}}\, e^{-\frac{1}{2}\boldsymbol{x}^\dagger \boldsymbol{\Sigma}_i^{-1}\boldsymbol{x}}, \quad i = 1, 2. \tag{5}$$

The global distribution of X is given by the mixture

$$f(\boldsymbol{x}) = P(\mathcal{C}_1)\, f(\boldsymbol{x}|\mathcal{C}_1) + P(\mathcal{C}_2)\, f(\boldsymbol{x}|\mathcal{C}_2).$$

Let $Y = \boldsymbol{w}^\dagger X$ be the projection of X into the direction defined by $\boldsymbol{w} \in \mathbb{R}^p$. Only the direction is important, so we can assume $\boldsymbol{w}$ to be

a vector of unit length. From basic statistics, the probability density function of Y is a mixture of Gaussians, i.e.,

$$f(y) = \sum_{k=1,2} \frac{P(\mathcal{C}_k)}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{y^2}{2\sigma_k^2}\right), \tag{6}$$

where $\sigma_k^2 = \boldsymbol{w}^\dagger \boldsymbol{\Sigma}_k \boldsymbol{w}$.

Now, we are interested in minimizing

$$D(\boldsymbol{w}) = \mathrm{E}\{|Y|\} = \mathrm{E}\{|\boldsymbol{w}^\dagger X|\} \tag{7}$$

over all possible projections defined by direction $\boldsymbol{w}$. Criterion $D(\boldsymbol{w})$ is quickly gaining popularity in the field of PCA for the following reason: standard PCA [23] aims to maximize the variance of Y which, for zero-mean data, is given by $\mathrm{E}\{Y^2\}$. Because it is raised to the square power, samples that are far apart from the nominal body of the data completely dominate the value of the variance. Therefore, standard PCA is very sensitive to outliers. An alternative is obtained by replacing Y^2 with $|Y|$, and it is in this way that we arrive at criterion D [18–20]. This variant is called L1-PCA because $D(\boldsymbol{w})$ is estimated in practice by the L1-norm of the vector that contains the samples of Y [18]. Again, we remark that [18,19] focus on *maximizing* $D(\boldsymbol{w})$, while we propose just the opposite.

The directions that solve the constrained optimization problem

$$\min_{\boldsymbol{w}} D(\boldsymbol{w}) \text{ subject to } \|\boldsymbol{w}\|^2 = 1 \tag{8}$$

verify

$$\nabla_{\boldsymbol{w}} D(\boldsymbol{w}) = \ell \, \nabla_{\boldsymbol{w}} \|\boldsymbol{w}\|^2, \tag{9}$$

where ℓ is a Lagrange multiplier and $\nabla_{\boldsymbol{w}}$ stands for the gradient with respect to $\boldsymbol{w}$. Under the Gaussian assumption (6), and after some algebraic derivations detailed in Appendix A, we obtain that

$$\nabla_{\mathbf{w}} D(\mathbf{w}) = \sqrt{\frac{2}{\pi}} \sum_{k=1}^{2} \frac{P(\mathcal{C}_k)}{\sigma_k} \boldsymbol{\Sigma}_k \mathbf{w}, \tag{10}$$

$$\nabla_{\boldsymbol{w}} \|\boldsymbol{w}\|^2 = 2\boldsymbol{w}, \tag{11}$$

$$\ell = \frac{1}{\sqrt{2\pi}} \sum_{k=1}^{2} P(\mathcal{C}_k)\sigma_k, \tag{12}$$

and hence the solutions of (9) satisfy

$$\sum_{k=1}^{2} \frac{P(\mathcal{C}_k)}{\sigma_k} \boldsymbol{\Sigma}_k \boldsymbol{w} = \left(\sum_{k=1}^{2} P(\mathcal{C}_k)\sigma_k\right) \boldsymbol{w}. \tag{13}$$

Invoking the whitening constraint (3), we get:

$$\left(\frac{\sigma_2 - \sigma_1}{\sigma_1}\right) P(\mathcal{C}_1) \boldsymbol{\Sigma}_1 \boldsymbol{w} = \left[\left(\sum_{k=1}^{2} P(\mathcal{C}_k)\sigma_k\right)\sigma_2 - 1\right] \boldsymbol{w}. \tag{14}$$

Then, by replacing the rightmost '1' with

$$1 = \boldsymbol{w}^\dagger \boldsymbol{\Sigma} \boldsymbol{w} = \sum_{k=1}^{2} P(\mathcal{C}_k)\sigma_k^2,$$

which follows from (2), and simplifying terms, the equation becomes:

$$(\sigma_2 - \sigma_1)\boldsymbol{\Sigma}_1 \boldsymbol{w} = (\sigma_2 - \sigma_1)\,\sigma_1^2\, \boldsymbol{w}. \tag{15}$$

Thus, apart from the solution $\sigma_1 = \sigma_2$ (which defines a maximum, see Appendix B), we find that:

Lemma 1. *Under the working assumption (6), the eigenvectors of* $\boldsymbol{\Sigma}_1$ *(or* $\boldsymbol{\Sigma}_2$*) are stationary points of (8).*

This result is complemented by the following one, which describes the minimizers:

Theorem 1. *For a p-dimensional random vector X distributed as a mixture of two multivariate Gaussian distributions with zero mean and different covariance matrices* $\boldsymbol{\Sigma}_1$ *and* $\boldsymbol{\Sigma}_2$ *verifying the whitening constraint (3), the minimizers of* $D(\boldsymbol{w})$ *are the eigenvectors associated with the maximum and minimum eigenvalues of* $\boldsymbol{\Sigma}_1$ *(or* $\boldsymbol{\Sigma}_2$*). The intermediate eigenvectors are saddle points, but are orthogonal to each other and can be found by a suitable optimization approach with orthogonal constraints.*

Proof and details are given in Appendix B. This result hence suggests an *unsupervised* approach to compute the FKT, based on the above L1-norm criterion. Furthermore, it endows the proposed criterion with discriminative properties in the case of equal-mean populations. Even though the theorem is derived by assuming Gaussian densities, it is still useful even when there are wide deviations from Gaussianity in the data distributions, as we will see in the experiments of Section 6.

5. Algorithm

Let us now propose a working algorithm for finding a set of appropriate projection vectors based on the above criterion. The algorithm is fully unsupervised, i.e., it does not require the labels of the data points.

5.1. Preprocessing

A few words about the whitening constraint (2) may be needed in the first place. To fulfill this condition, we will often require in practice a pre-processing of the data. Specifically, given a 'colored' (i.e., non-white) random vector $X_c \in \mathbb{R}^P$, assumed to have zero mean, whitened data X can be obtained, for example, as follows [24]:

$$X = \boldsymbol{\Gamma}^{-1/2} \boldsymbol{V}^\dagger X_c,$$

where $\boldsymbol{V}$ is the matrix whose columns are the eigenvectors of $\mathrm{E}\{X_c X_c^\dagger\}$, and $\boldsymbol{\Gamma}$ is the diagonal matrix of its eigenvalues (note that there exist other whitening approaches that are equally valid [24]). It is straightforward to check that, as desired, $\boldsymbol{\Sigma} = \mathrm{E}\{X X^\dagger\} = \boldsymbol{I}$.

5.2. Algorithm for joint L1-norm minimization

The maximization of the L1-norm criterion $D(\boldsymbol{w})$ defined in (7)–(8) has already been studied in a number of recent works [18,19,25]. Unfortunately, because of how they have been designed, none of these ad-hoc maximization approaches can be turned into a minimization algorithm. Therefore, we opt here for a gradient-based approach.

As the eigenvectors of the class covariance matrices are always orthogonal, they can be determined by successively minimizing $D(\boldsymbol{w})$ under the constraint that the direction obtained in the current minimization is orthogonal to the previously computed ones (see Appendix B). However, this simple deflation approach has the disadvantage of accumulating estimation errors along successively calculated directions. To avoid these drawbacks, we consider the cost function

$$J(\boldsymbol{W}) = \sum_{i=1}^{p} D(\boldsymbol{w}_i),$$

where $\boldsymbol{W}$ is the matrix $[\boldsymbol{w}_{:,1}\, \boldsymbol{w}_{:,2} \cdots \boldsymbol{w}_{:,p}]$ containing the projection vectors, where $\boldsymbol{w}_{:,n} = \boldsymbol{w}_n$ denotes its n-th column. We are interested in its minimization with orthogonality constrains

$$\min_{\boldsymbol{W}} J(\boldsymbol{W})$$

$$\text{s.t. } \boldsymbol{W}^\dagger \boldsymbol{W} = \boldsymbol{I}. \tag{16}$$

Following classical results of optimization over the set of orthogonal matrices [26], we adopt a minimization approach based on gradient-descent in the Stiefel manifold. As justified in Appendix D, this approach leads to the following multiplicative update scheme, which preserves the orthogonality constraint during iterations:

$$\boldsymbol{W}_{n+1} = \boldsymbol{U}_n \boldsymbol{W}_n, \tag{17}$$

where $\boldsymbol{U}_n = \exp(\boldsymbol{S}_n) = \boldsymbol{I} + \boldsymbol{S}_n + \frac{1}{2!}\boldsymbol{S}_n^2 + \dots$ Choosing a skew-symmetric matrix $\boldsymbol{S}_n$, i.e., $\boldsymbol{S}_n = -\boldsymbol{S}_n^\dagger$, guarantees the orthogonality of matrix $\boldsymbol{U}_n$, and thus that of $\boldsymbol{W}_{n+1}$. Apart from fulfilling this condition, matrix $\boldsymbol{S}_n$ is closely related to the gradient of J, allowing update (17) to perform gradient descent, as detailed next.

The algorithm can be described as follows. Starting from any initial orthogonal matrix $\boldsymbol{W}_0$, repeat the following three steps for $n = 0, 1, 2, \dots$ until convergence [26]:

1: Set

$$\boldsymbol{S}_n = \partial J(\boldsymbol{W}_n)\boldsymbol{W}_n^\dagger - \boldsymbol{W}_n \partial J(\boldsymbol{W}_n)^\dagger$$

where $\partial J(\boldsymbol{W})$ is the matrix of partial derivatives of J with respect to the elements of $\boldsymbol{W}$, i.e.,

$$(\partial J(\boldsymbol{W}))_{ij} = \frac{\partial J(\boldsymbol{W})}{\partial \boldsymbol{W}_{ij}} \tag{18}$$

(this equation will be detailed below).

2: Define $\boldsymbol{U}_n = \exp(-\eta\, \boldsymbol{S}_n)$ for $\eta \in \mathbb{R}^+$ small enough.

3: Update $\boldsymbol{W}_{n+1} = \boldsymbol{U}_n \boldsymbol{W}_n$.

It still remains to give a formula for $\partial J(\boldsymbol{W})$ in Eq. (18). Subderivatives, or subgradients, generalize the notion of derivative to non-differentiable convex functions [27]. As the subderivative of the absolute value is the sign function, it is easily found that the nth column of $\partial J(\boldsymbol{W})$ equals

$$\frac{\partial}{\partial \boldsymbol{w}_n} D(\boldsymbol{w}_n) = \mathrm{E}\left\{X \operatorname{sgn}\left(\boldsymbol{w}_n^\dagger X\right)\right\},$$

with D defined in (7). As a simple illustration, given a $p \times q$ data matrix `M_x` containing q observed samples of X, $\partial J(\boldsymbol{W})$ can be evaluated by the MATLAB® command `M_x*sign(W'*M_x)'/q;`. Observe that this calculation does not require any knowledge of the class data labels.

5.2.1. Interpretation of the method

The above algorithm can be easily viewed as a gradient descent approach. To see this, we note that, for small η,

$$\boldsymbol{U}_n = \exp(-\eta\, \boldsymbol{S}_n) \approx \boldsymbol{I} - \eta \boldsymbol{S}_n \tag{18}$$

and therefore

$$\boldsymbol{W}_{n+1} = \boldsymbol{U}_n \boldsymbol{W}_n \approx \boldsymbol{W}_n - \eta\, \boldsymbol{S}_n \boldsymbol{W}_n. \tag{19}$$

Bounds on approximation (18) are given in Appendix C, and show its pertinence for sufficiently small η. Interestingly, the term

$$\begin{aligned} \nabla J(\boldsymbol{W}_n) &= \boldsymbol{S}_n \boldsymbol{W}_n \\ &= \partial J(\boldsymbol{W}_n)\, \boldsymbol{W}_n^\dagger \boldsymbol{W}_n - \boldsymbol{W}_n \partial J(\boldsymbol{W}_n)^\dagger \boldsymbol{W}_n \\ &= \partial J(\boldsymbol{W}_n) - \boldsymbol{W}_n\, \partial J(\boldsymbol{W}_n)^\dagger\, \boldsymbol{W}_n \end{aligned} \tag{20}$$

is, up to an irrelevant scale factor, the gradient of J in the set of orthogonal matrices, i.e., the projection of the gradient of J on the tangent space of the Stiefel manifold. Details are given in Appendix D. This relation allows us to interpret (19) as an approximate gradient rule, i.e.,

$$\Delta \boldsymbol{W} = \boldsymbol{W}_{n+1} - \boldsymbol{W}_n \approx -\eta \nabla J(\boldsymbol{W}_n).$$

Now, consider the first-order Taylor expansion of J

$$J(\boldsymbol{W} + \Delta \boldsymbol{W}) = J(\boldsymbol{W}) + \langle \partial J(\boldsymbol{W}) | \Delta \boldsymbol{W} \rangle + \dots,$$

where $\langle \partial J(\boldsymbol{W}) | \Delta \boldsymbol{W} \rangle = \operatorname{trace}(\partial J(\boldsymbol{W})^\dagger \Delta \boldsymbol{W})$. By setting, as before,

$$\Delta \boldsymbol{W} = -\eta \nabla J(\boldsymbol{W}),$$

some algebra shows that

$$\langle \partial J(\boldsymbol{W}) | \Delta \boldsymbol{W} \rangle = -\frac{\eta}{2} \langle \nabla J(\boldsymbol{W}) | \nabla J(\boldsymbol{W}) \rangle,$$

which is always negative for sufficiently small adaption step η (otherwise, the first-order Taylor expansion is no longer valid) and, therefore, $J(\boldsymbol{W})$ decreases with every update as desired. Note finally that, if $\boldsymbol{W}$ contains the eigenvectors of $\boldsymbol{\Sigma}_1$ (or $\boldsymbol{\Sigma}_2$) in its columns, it follows from (9)–(12) that $\partial J(\boldsymbol{W}) = \boldsymbol{W}\boldsymbol{\Lambda}$, where $\boldsymbol{\Lambda}$ is a diagonal matrix containing twice the Lagrange multipliers (12). Therefore, matrix

$$\boldsymbol{S} = \partial J(\boldsymbol{W})\boldsymbol{W}^\dagger - \boldsymbol{W}\partial J(\boldsymbol{W})^\dagger$$

vanishes and the iteration stops.

6. Experimental assessment

Experiments are next performed in a variety of conditions. Tests are applied to both synthetic and real electroencephalographic (EEG) data sets.

6.1. Bivariate Gaussian data

Let us first consider a mixture in a bidimensional space (i.e., $p = 2$) of two equiprobable Gaussian classes with zero-means and respective covariances

$$\boldsymbol{\Sigma}_1 = \begin{pmatrix} 1 & 0.68 \\ 0.68 & 1 \end{pmatrix} \text{ and } \boldsymbol{\Sigma}_2 = \begin{pmatrix} 1 & -0.68 \\ -0.68 & 1 \end{pmatrix}. \tag{21}$$

$\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ fulfill the whitening condition (2) and share the same eigenvectors, i.e.,

$$\begin{aligned} \boldsymbol{v}_1 &= \left(\frac{1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)^\dagger \approx (0.71, 0.71)^\dagger, \quad \boldsymbol{v}_2 \\ &= \left(\frac{-1}{\sqrt{2}}, \frac{1}{\sqrt{2}}\right)^\dagger \approx (-0.71, 0.71)^\dagger. \end{aligned} \tag{22}$$

We draw 50 samples from each class (100 samples in total), whose scatter plot is shown in Fig. 1. The lines through $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$ are plotted in dashed blue; as recalled in Section 3, classes $\mathcal{C}_1$ and $\mathcal{C}_2$ can be well reconstructed in the line spanned by one of these eigenvectors, and not so well in the line spanned by the other. Here, 'well' means that the average squared distance of the points to the line is minimized. We also draw in red the lines pointing in the direction of

$$\boldsymbol{w}_1 \approx (0.73, 0.69)^\dagger, \quad \boldsymbol{w}_2 \approx (-0.69, 0.73)^\dagger. \tag{23}$$

These are the minimizers of the L1-norm calculated by the unsupervised algorithm presented in Section 5. As $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$ are estimates of $\boldsymbol{v}_1$ and $\boldsymbol{v}_2$, we can cluster the observations $\boldsymbol{x}$ into two groups, say A and B, based on their closeness to the subspaces spanned by $\boldsymbol{w}_1$ and $\boldsymbol{w}_2$:

$$\begin{aligned} &\text{if } \left\|\boldsymbol{x} - \boldsymbol{w}_1(\boldsymbol{w}_1^\dagger \boldsymbol{x})\right\| < \left\|\boldsymbol{x} - \boldsymbol{w}_2(\boldsymbol{w}_2^\dagger \boldsymbol{x})\right\|, \\ &\quad \text{assign } \boldsymbol{x} \in A \text{ otherwise } \boldsymbol{x} \in B. \end{aligned} \tag{24}$$

By applying this rule, we obtain the confusion matrix shown in Table 2. There is one cluster composed of 35 samples of class 1 and only 13 of class 2, and a second group with 15 instances of class 1 and 37 of class 2. We see that most data points of the same class lie together, which is what one would expect from an unsupervised method. To compute the accuracy, we sum the values on the diagonal of the confusion matrix and divide by the number

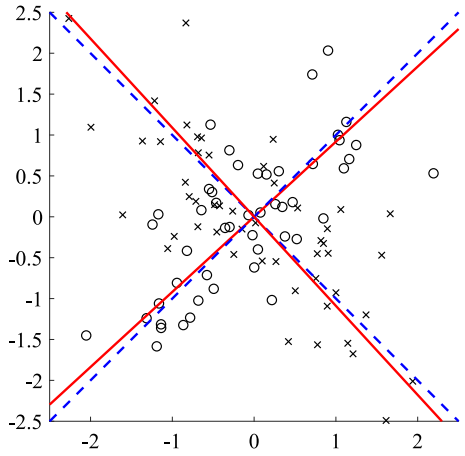


Fig. 1. Scatter plot of the observations of the two classes classes ('crosses' and 'circles'). In dashed blue, we show the lines through the eigenvectors of the class covariance matrices. Red lines point to the projection directions found by the unsupervised algorithm in Section 5. We observe that 'red' axes are rotated through an angle of 1.61° with respect to the 'blue' ones. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 2
Confusion matrix obtained after applying the allocation rule (24).

		Actual class		
		$\mathcal{C}_1$	$\mathcal{C}_2$	Total
Assigned cluster	A	35	13	48
	B	15	37	52
	Total	50	50	100

of samples: overall, we have $35 + 37 = 72$ correctly clustered data points out of 100, implying that the method provides an accuracy of 72%.

Let us see it another way. For any given observed data value $\boldsymbol{x}$, (24) is equivalent to

$$|\boldsymbol{w}_1^\dagger \boldsymbol{x}| > |\boldsymbol{w}_2^\dagger \boldsymbol{x}|. \tag{25}$$

Fig. 2(a) shows that inequality (25) usually holds true for the elements of $\mathcal{C}_1$ as the orange line is usually above the green one. For the elements of $\mathcal{C}_2$, as seen in Fig. 2b, it is just the opposite. To quantify this inverse relationship, we compute Pearson's correlation coefficient of the absolute projections, defined as

$$\rho_{ij} = \frac{\mathrm{cov}(Z_i, Z_j)}{\sigma_i \sigma_j}$$

where $\mathrm{cov}(\cdot,\cdot)$ denotes the covariance of its input variables and σ_i is the standard deviation of $Z_i = |\boldsymbol{w}_i^\dagger X|$. The value of Pearson's correlation coefficient ρ_{12} between $Z_1 = |\boldsymbol{w}_1^\dagger X|$ and $Z_2 = |\boldsymbol{w}_2^\dagger X|$ becomes negative,

$$\rho_{12} = -0.254,$$

indicating that, on average, the magnitudes of the projected points for one class and for the other show opposite behavior.

6.1.1. Multivariate Gaussian and non-Gaussian data

When applied to p-dimensional data points from two equiprobable classes, the algorithm in Section 5 finds p projection vectors $\boldsymbol{w}_1, \ldots, \boldsymbol{w}_p \in \mathbb{R}^p$. To determine which of them correspond to the two most discriminant directions, one can choose the minimizers of $D(\boldsymbol{w})$ as in Theorem 1. We further introduce a slight refinement that experimentally improves the robustness of the classification against errors in the estimation of the eigenspace due to the finite sample size. Let us arrange these vectors so that: (i) $\boldsymbol{w}_1$ is the global minimizer of the L1-norm criterion, and (ii) among all remaining directions $\boldsymbol{w}_i$, $i > 1$, Pearson's correlation coefficient between $|\boldsymbol{w}_1^\dagger X|$ and $|\boldsymbol{w}_p^\dagger X|$ is the most negative, $\rho_{1p} < \rho_{1i}$ for $i \neq p$. As in the previous experiment, $\boldsymbol{w}_1$ and $\boldsymbol{w}_p$ are expected to represent different classes. Then, inspired by (24), we adopt the rule:

$$\text{Assign } \boldsymbol{x} \text{ to } A \text{ if: } \left\|\boldsymbol{x} - \boldsymbol{w}_1(\boldsymbol{w}_1^\dagger \boldsymbol{x})\right\| < \left\|\boldsymbol{x} - \boldsymbol{w}_p(\boldsymbol{w}_p^\dagger \boldsymbol{x})\right\|$$

otherwise, assign $\boldsymbol{x}$ to B. Fig. 3 shows the accuracy of this fully unsupervised classification approach, calculated as in the previous experiment, when tested on different data distributions and values of p. In each simulation, the covariance matrices are generated at random, and the data are whitened as in Section 5.1 to fulfill condition (2). In addition, we draw $N = 50\,p$ samples per each of the two classes, using the algorithms in [28,29] for generating multivariate non-Gaussian data with the specified covariances. These algorithms, widely used in robustness analysis, nonlinearly transform multivariate random Gaussian variables in a way that allows us to fix at will the mean, variance, skewness and kurtosis of the resulting marginal distributions. Specifically, we generate zero-mean, unit variance and zero-skew marginal data. Nevertheless, to explore different scenarios, we consider different values of excess kurtosis κ of the marginal data. Recall that the excess kurtosis is defined as the 4th-order central moment of the standardized (zero-mean, unit-variance) data minus three. In this experiment, κ is varied between -1 (which corresponds to a sub-Gaussian density) to 5 (highly super-Gaussian distribution), passing through 0 (Gaussian variable). Hence, we can test the performance of the algorithm in Section 5 when the assumption (5) for normality of data is not fulfilled. A notable feature is that, as seen in Fig. 3, the performance of the algorithm increases with the dimensionality of the input representation, which could be explained by the fact that it is generally easier to discriminate between classes in a feature space of higher dimensions.

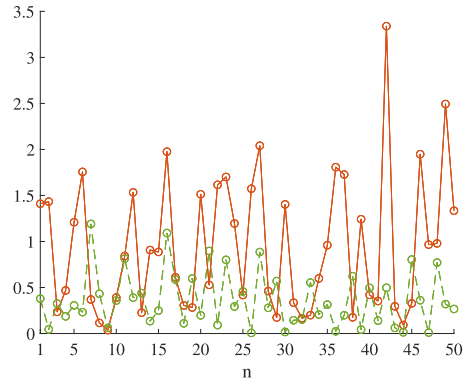
Table 3
Mean number of iterations (MNI) before the convergence of the algorithm as a function of the dimensionality p of the data, averaged over all distributions.

p	2	5	10	15	20	25	30
MNI	6.6	54.5	193.8	349.7	514.2	686.0	813.1
$\frac{\text{MNI}}{p}$	3.3	10.9	19.4	23.3	25.7	27.4	27.1

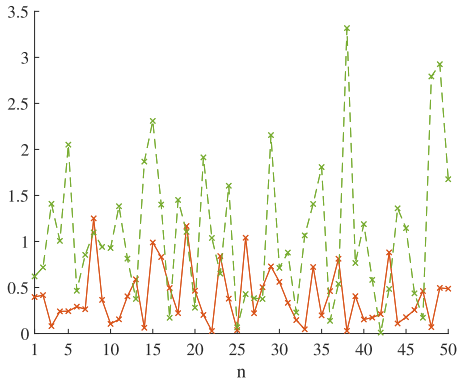
Furthermore, to speed up the algorithm, we have chosen in each iteration the step size η that gives the maximum reduction in the value of the cost function. Simple line-search algorithms, such as the golden-section search method, can be used to solve this problem [30]. Fig. 4 illustrates the convergence of the algorithm for the case of $p = 15$-dimensional data, suggesting that it is roughly independent of the value of the excess kurtosis of the data. In all cases, the algorithm stops when $\|\boldsymbol{W}_{n+1} - \boldsymbol{W}_n\| < 10^{-4}$, where $\boldsymbol{W}_n$ the value of matrix $\boldsymbol{W}$ after the n-th iteration. Table 3 shows the mean number of iterations, averaged over all distributions, before the convergence of the algorithm.

Additionally, L1-norm criteria are also expected to exhibit robustness against large outliers. To test this property, we repeat the experiment with the difference that the data points are now corrupted by replacing 10 per cent of the data samples, at randomly chosen time instants, by Gaussian noise realizations with identity covariance matrix and mean $\boldsymbol{\mu}_{\text{outliers}} = [10, 10, \ldots, 10]^\dagger$, which denotes a p-dimensional vector with all elements equal to 10.

In this new experiment, we have to take into account that the usual covariance estimate is very sensitive to the presence of outliers in the data set and, therefore, the whitening pre-processing, which is ultimately based on the eigendecomposition of that covariance matrix, inherits this sensitivity. To prevent this from affecting the experiment, whitening is performed by using a Fast-MCD robust estimator of the data covariance [31]. The new results



(a) Values of $|\boldsymbol{w}_1^\dagger \boldsymbol{x}_n|$ (continuous orange line) and and $|\boldsymbol{w}_2^\dagger \boldsymbol{x}_n|$ (dashed green line) for the 50 data points $\boldsymbol{x}_n$ in class $\mathcal{C}_1$.



(b) Values of $|\boldsymbol{w}_1^\dagger \boldsymbol{x}_n|$ (continuous orange line) and and $|\boldsymbol{w}_2^\dagger \boldsymbol{x}_n|$ (dashed green line) for the 50 data points $\boldsymbol{x}_n$ in class $\mathcal{C}_2$.

Fig. 2. Absolute value of the projected data points from (a) class $\mathcal{C}_1$ and (b) class $\mathcal{C}_2$.

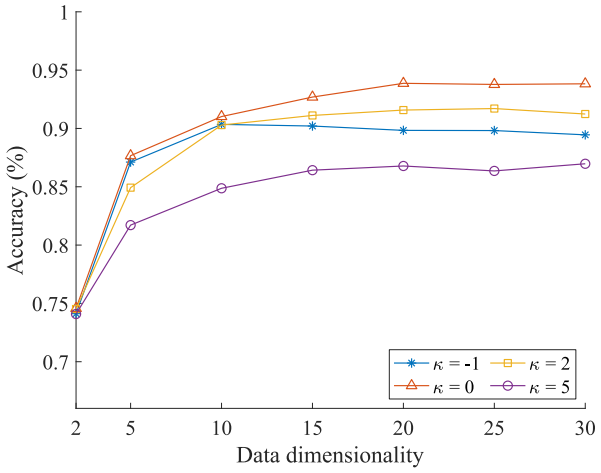


Fig. 3. Accuracy of the proposed method, as the dimensionality p of the data increases, for distributions with different excess kurtoses κ (e.g. $\kappa = -1.2$ corresponds to uniformly distributed marginals, $\kappa = 0$ to the Gaussian distribution or $\kappa = 3$ gives the Laplace distribution). Each curve has been obtained by averaging over 100 independent experiments.

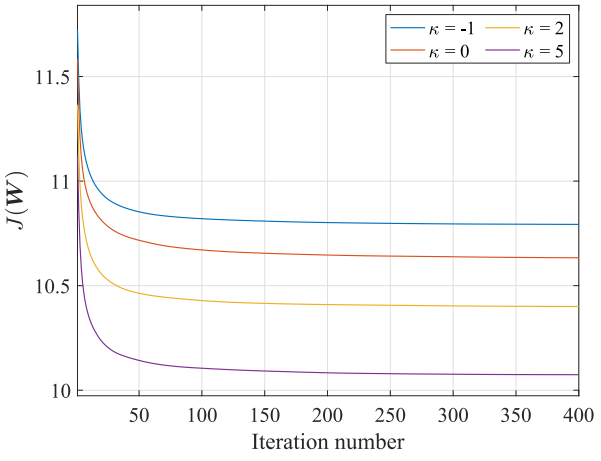


Fig. 4. Convergence of the algorithm as a function of the iteration number for distributions with different excess kurtoses κ and $p = 15$-dimensional data. The curves are obtained by averaging 100 independent experiments.

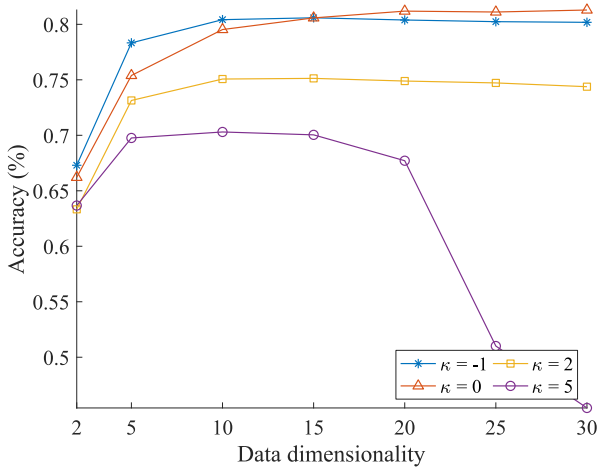


Fig. 5. Accuracy of the proposed method, for distributions with different excess kurtoses κ, when 10 per cent of the data samples are replaced, at randomly chosen time instants, by large outliers. The data covariance matrix, which is necessary for performing the whitening pre-processing, has been estimated by using a robust method.

are represented in Fig. 5. Observe that the most leptokurtic distribution, that with $\kappa = 5$, seems to be severely affected by the presence of outliers. To confirm if this is true, an additional simulation is performed in which the data were whitened before outliers were added, i.e., the covariance matrix was calculated from the outlier-free observations. Fig. 6 shows that the improvement obtained with respect to the previous case is remarkable. We conclude that it is the whitened pre-processing step, which requires estimating a covariance matrix, which actually limits the ability of the proposed technique to fight against outliers. Thus, to fully exploit the capabilities of the L1-norm algorithm, it must be combined with a robust covariance estimator that guarantees that the pre-whitening step is also resistant to outliers. This is not actually surprising, as the traditional FKT also requires a robust estimation of the class covariance matrices.

6.2. Real electroencephalographic (EEG) data

In motor imagery-based brain computer interfaces (BCI's), the user imagines a limb moving and the system tries to identify the imagined movement by analyzing the EEG data recorded dur-

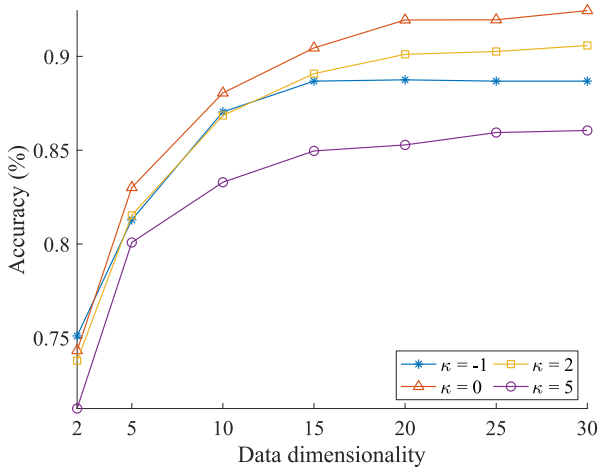


Fig. 6. Accuracy of the proposed method, for distributions with different excess kurtoses κ, when 10% of the data samples are replaced, at randomly chosen time instants, by large outliers. The whitening transformation is performed by the true (outlier-free) covariance matrix of the observations.

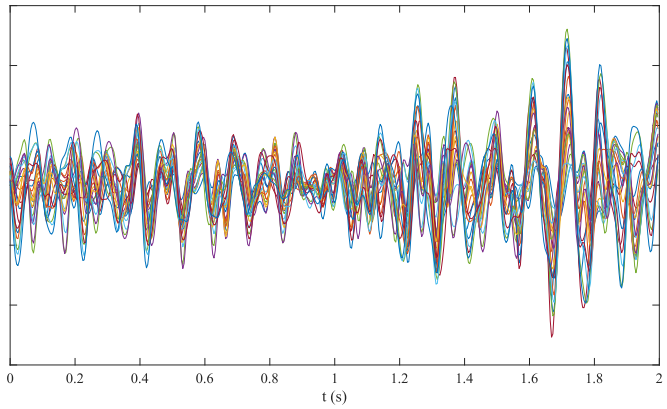


Fig. 7. Butterfly plot of 22-channel EEG recorded while subject number 1 imagines movements of tongue.

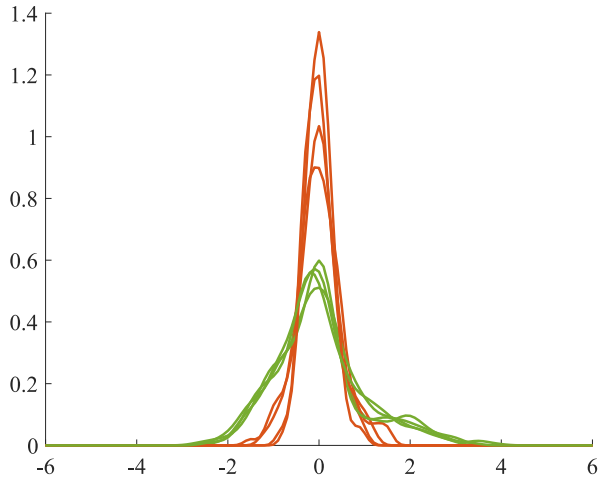


Fig. 8. Density functions (produced with a kernel density estimation method, with Gaussian kernel and Silverman's optimal bandwidth) of the scalar projections of the data from several projected 'left hand' (orange curves) and 'feet' trials (green curves) from user 1. The projection direction is that which minimizes the L1-norm-based objective function, calculated by the algorithm in Section 5 when using as input all 'left-hand' and 'feet' trials of user 1. The difference between the respective variances is apparent. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Table 4
Accuracy in the discrimination between pairs of imagined movements (L = left hand, R = right hand, F = feet, T = tongue). Results are shown for the nine users in the database (u1,. . . , u9). Last column gives the accuracy per user averaged over all possible pairs of movements. The last row is the average of all the previous rows.

User	L-R	L-F	L-T	R-F	R-T	F-T	avg
u1	0.66	0.89	0.91	0.93	0.92	0.52	0.84
u2	0.52	0.72	0.6	0.68	0.54	0.65	0.64
u3	0.87	0.68	0.69	0.86	0.86	0.54	0.73
u4	0.57	0.7	0.61	0.63	0.65	0.55	0.63
u5	0.53	0.55	0.6	0.55	0.57	0.54	0.56
u6	0.52	0.64	0.56	0.53	0.54	0.53	0.56
u7	0.59	0.73	0.74	0.89	0.89	0.69	0.79
u8	0.77	0.65	0.87	0.59	0.75	0.71	0.72
u9	0.78	0.86	0.88	0.55	0.7	0.76	0.75
avg	0.65	0.71	0.72	0.69	0.71	0.61	

ing the experiment [32,33]. The dataset 2a from the BCI competition IV comprises a number of trials (repetitions) of some simple limb (left hand, right hand, feet or tongue) motor-imagery movements [34–36]. In each recording session, $p = 22$-channel EEG signals are measured at a sample rate of 250 Hz from volunteers performing the desired imagery tasks. As usual in BCI signal processing, the EEG data are bandpass filtered to $8 - 30$ Hz. This preprocessing ensures that the data are zero-mean and, by central limit arguments, also allows us to support the hypothesis of Gaussianity for long filters.

Each imagined action lasts for about three seconds, but only the final two of them are kept in our experiment to avoid the initial transient effects. For illustration, one of these two-second intervals is shown in Fig. 7. We concatenate all trials of the same imagined movement into a single $22 \times 30\,000$ data matrix, and the algorithm in Section 5 is fed with pairs of matrices of distinct imagined movements. As an example, Fig. 8 depicts the density functions of the scalar projection of some data points, from trials of two distinct imaginary tasks, onto the direction that minimizes the L1 criterion: the differences in variance between the two classes are apparent even to the naked eye. Best results are obtained for data filtered in the band between 12 and 30 Hz (upper α and β bands), as well as pre-processed with the method in [37] to reduce the inherent nonstationarity of the EEG.

Next, for the set of data points of each trial, we retain the scalar projection in the direction of minimum L1-norm and the two projections which are most correlated with the first one, as well as the three projections with the lowest correlation, in a similar way as we have done before in Section 6.1.1. Table 4 shows how well a given imagined movement is classified simply by comparing the total variances of these two groups of three projections. As the trials are actually time-series, and not just a point in a p-dimensional space, comparing variances is easier to do and a feasible criterion. Note that the total variance in each projected subspace is measured by the trace of the covariance matrix of the projected data.

Accuracy is measured for the nine voluntaries in the database and considering all the possible combination of imagined tasks (L-R: left hand-right hand, L-F: left hand-feet, and so on). For example, a high degree (93%) of accuracy in discriminating between 'right hand' and 'feet' imagined movements has been obtained for user 1, achieved in a completely unsupervised fashion, but that accuracy reduces to 52% for the same user and the pair 'feet-tongue'. There is a great variability between users and pairs of movements, nevertheless, averaged over all users, we can discriminate between 'left-hand' and 'feet', 'left-hand' and 'tongue', and 'right-hand' and 'tongue' movements in more than 70% of the cases.

7. Conclusions

Projecting whitened data onto the few dimensions that minimize the absolute value of the projected data points can perform the FKT in a fully unsupervised fashion, sparing the need for training data. This connection between the L1-norm and the FKT had previously gone unnoticed, and endows L1-criteria with discriminative properties in binary classification scenarios, opening new lines of research in the area of L1-PCA. A working iterative algorithm based on gradient-descent in the Stiefel manifold is also put forward to perform L1-norm minimization with orthogonal constraints. Even though our theoretical analysis assumes the normality of the data, numerical experiments show that a good performance can be achieved when this assumption is not fulfilled. Further theoretical research should explore this extension to scenarios with non-Gaussian data.

Declaration of Competing Interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix A. Proof of eqn. (13)

The cost function is defined as follows:

$$D(\boldsymbol{w}) = \mathrm{E}\{|Y|\} = \int_{-\infty}^{\infty} |y|\, f(y)\, \mathrm{d}y = \int_{0}^{\infty} y\, f(y)\, \mathrm{d}y - \int_{-\infty}^{0} y\, f(y)\, \mathrm{d}y, \tag{A.1}$$

and invoking the zero mean assumption, i.e.,

$$\mathrm{E}\{Y\} = \int_{0}^{\infty} y\, f(y)\, \mathrm{d}y + \int_{-\infty}^{0} y\, f(y)\, \mathrm{d}y = 0 \;\Rightarrow\; \int_{-\infty}^{0} y\, f(y)\, \mathrm{d}y = -\int_{0}^{\infty} y\, f(y)\, \mathrm{d}y,$$

we readily get

$$D(\boldsymbol{w}) = \mathrm{E}\{|Y|\} = 2\int_{0}^{\infty} y\, f(y)\, \mathrm{d}y.$$

Under the Gaussian assumption (6), i.e.,

$$f(y) = \sum_{k=1,2} \frac{P(\mathcal{C}_k)}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{y^2}{2\sigma_k^2}\right),$$

where

$$\sigma_k^2 = \boldsymbol{w}^\dagger \boldsymbol{\Sigma}_k \boldsymbol{w}, \tag{A.2}$$

the cost function can be worked out as:

$$D(\boldsymbol{w}) = 2\sum_{k=1}^{2} P(\mathcal{C}_k) \int_{0}^{\infty} \frac{y}{\sqrt{2\pi\sigma_k^2}} \exp\left(-\frac{y^2}{2\sigma_k^2}\right) \mathrm{d}y = \sqrt{\frac{2}{\pi}} \sum_{k=1}^{2} P(\mathcal{C}_k)\, \sigma_k, \tag{A.3}$$

where the second equality is readily obtained by using the identity [38]:

$$\int y\, e^{-\frac{y^2}{2\sigma^2}} \mathrm{d}y = -\sigma^2\, e^{-\frac{y^2}{2\sigma^2}} + \text{constant of integration.}$$

The stationary points of the constrained optimization problem (8) verify

$$\nabla_{\boldsymbol{w}} D(\boldsymbol{w}) = \ell\, \nabla_{\boldsymbol{w}} \|\boldsymbol{w}\|^2. \tag{A.4}$$

where ℓ is a Lagrange multiplier and $\nabla_{\boldsymbol{w}}$ stands for the gradient with respect to $\boldsymbol{w}$. We see that (A.3) is a function of σ_1 and σ_2. It is easier to calculate $\nabla_{\boldsymbol{w}}\sigma_k$ by noticing that

$$\nabla_{\boldsymbol{w}}\sigma_k^2 = 2\sigma_k \nabla_{\boldsymbol{w}}\sigma_k,$$

and, as follows from (A.2), that

$$\nabla_{\boldsymbol{w}}\sigma_k^2 = \nabla_{\boldsymbol{w}}\left(\boldsymbol{w}^\dagger \boldsymbol{\Sigma}_k \boldsymbol{w}\right) = 2\,\boldsymbol{\Sigma}_k \boldsymbol{w}.$$

Combining both formulas, we readily get $\nabla_{\boldsymbol{w}}\sigma_k = \nabla_{\boldsymbol{w}}\sigma_k^2/(2\sigma_k) = \boldsymbol{\Sigma}_k\boldsymbol{w}/\sigma_k$. Replacing this result in the calculation of the gradient of (A.3), it follows that

$$\nabla_{\boldsymbol{w}} D(\boldsymbol{w}) = \sqrt{\frac{2}{\pi}} \sum_{k=1}^{2} \frac{P(\mathcal{C}_k)}{\sigma_k}\, \boldsymbol{\Sigma}_k \boldsymbol{w}. \tag{A.5}$$

Similarly,

$$\nabla_{\boldsymbol{w}} \|\boldsymbol{w}\|^2 = \nabla_{\boldsymbol{w}}\left(\boldsymbol{w}^\dagger \boldsymbol{w}\right) = 2\boldsymbol{w}. \tag{A.6}$$

Therefore, (A.4) becomes

$$\sqrt{\frac{2}{\pi}} \sum_{k=1}^{2} \frac{P(\mathcal{C}_k)}{\sigma_k}\, \boldsymbol{\Sigma}_k \boldsymbol{w} = 2\,\ell\,\boldsymbol{w}. \tag{A.7}$$

The value of ℓ can be obtained by pre-multiplying (A.7) by $\boldsymbol{w}^\dagger$, after which we use (A.2) as well as $\boldsymbol{w}^\dagger\boldsymbol{w} = 1$. By so doing, we finally get:

$$\ell = \frac{1}{\sqrt{2\pi}} \sum_{k=1}^{2} P(\mathcal{C}_k)\sigma_k. \tag{A.8}$$

Appendix B. Proof of Theorem 1

Let us study whether the eigenvectors of $\boldsymbol{\Sigma}_1$ and $\boldsymbol{\Sigma}_2$ correspond to maxima, minima or saddle points of the L1-norm objective function. We start by calculating the Hessian (matrix of second order partial derivatives) of $\mathrm{E}\{|Y|\}$. From (A.5), and after some algebra, this Hessian is easily found to be:

$$\Delta_{\boldsymbol{w}}^2 \mathrm{E}\{|Y|\} = \sqrt{\frac{2}{\pi}} \sum_{k=1}^{2} \frac{P(\mathcal{C}_k)}{\sigma_k} \left[\boldsymbol{\Sigma}_k - \frac{1}{\sigma_k^2}\boldsymbol{\Sigma}_k \boldsymbol{w}\,(\boldsymbol{\Sigma}_k \boldsymbol{w})^\dagger\right] = \sqrt{\frac{2}{\pi}} \sum_{k=1}^{2} \frac{P(\mathcal{C}_k)}{\sigma_k} \left[\boldsymbol{\Sigma}_k - \sigma_k^2 \boldsymbol{w}\boldsymbol{w}^\dagger\right]. \tag{B.1}$$

where the second equality follows from $\boldsymbol{\Sigma}_k\boldsymbol{w} = \sigma_k^2\boldsymbol{w}$. Similarly, the Hessian matrix of the constraint $\|\boldsymbol{w}\|^2 = 1$ reads

$$\Delta_{\boldsymbol{w}}^2 \|\boldsymbol{w}\|^2 = 2\boldsymbol{I}. \tag{B.2}$$

Finally, the Hessian of the Lagrangian equals

$$\Delta_{\boldsymbol{w}}^2 L = \Delta_{\boldsymbol{w}}^2 \mathrm{E}\{|Y|\} - \ell\, \Delta_{\boldsymbol{w}}^2 \|\boldsymbol{w}\|^2, \tag{B.3}$$

where ℓ is the Lagrange multiplier. Then, note the following result in [39, Chap. 20], which we rewrite here in our own notation:

Theorem 2. *Let $\boldsymbol{w}$ be a critical point (maximizer, minimizer or saddle point) of $\mathrm{E}\{|Y|\}$ subject to $\|\boldsymbol{w}\|^2 = 1$. If, for all unit-length vector $\boldsymbol{z}$ such that*

$$\boldsymbol{z}^\dagger \nabla_{\boldsymbol{w}} \|\boldsymbol{w}\|^2 = 2\boldsymbol{z}^\dagger\boldsymbol{w} = 0, \tag{B.4}$$

it holds that

$$\boldsymbol{z}^\dagger (\Delta_{\boldsymbol{w}}^2 L)\boldsymbol{z} > 0, \tag{B.5}$$

then $\boldsymbol{w}$ is a local minimizer. For a local maximizer, the above condition becomes $\boldsymbol{z}^\dagger(\Delta_{\boldsymbol{w}}^2 L)\boldsymbol{z} < 0$.

By using (A.8), we get

$$\boldsymbol{z}^\dagger \Delta_{\boldsymbol{w}}^2 L\, \boldsymbol{z} = \sqrt{\frac{2}{\pi}} \left(\sum_{k=1}^{2} \frac{s_k^2 - \sigma_k^2}{\sigma_k} P(\mathcal{C}_k)\right), \tag{B.6}$$

where $\sigma_k^2 = \boldsymbol{w}^\dagger \boldsymbol{\Sigma}_k \boldsymbol{w}$ and $s_k^2 = \boldsymbol{z}^\dagger \boldsymbol{\Sigma}_k \boldsymbol{z}$. Let us analyze the term:

$$\frac{s_2^2 - \sigma_2^2}{\sigma_2} P(\mathcal{C}_2). \tag{B.7}$$

On the one hand, the whitening condition (2) allows us to write

$$1 = \boldsymbol{z}^\dagger \boldsymbol{\Sigma} \boldsymbol{z} = \sum_{k=1}^{2} P(\mathcal{C}_k)\, s_k^2 \;\Rightarrow\; s_2^2 P(\mathcal{C}_2) = 1 - P(\mathcal{C}_1)\, s_1^2 \tag{B.8}$$

and, by the same token,

$$\sigma_2^2 P(\mathcal{C}_2) = 1 - P(\mathcal{C}_1)\, \sigma_1^2. \tag{B.9}$$

Invoking these results, we get

$$\frac{s_2^2 - \sigma_2^2}{\sigma_2} P(\mathcal{C}_2) = -\frac{(s_1^2 - \sigma_1^2)}{\sigma_2} P(\mathcal{C}_1). \tag{B.10}$$

Hence, substituting in (B.6), it follows that

$$\boldsymbol{z}^\dagger \Delta_{\boldsymbol{w}}^2 L \boldsymbol{z} = \sqrt{\frac{2}{\pi}} (s_1^2 - \sigma_1^2) \Big(\frac{1}{\sigma_1} - \frac{1}{\sigma_2} \Big) P(\mathcal{C}_1). \tag{B.11}$$

Let $\boldsymbol{v}_1, \ldots, \boldsymbol{v}_p$ be the eigenvectors of $\boldsymbol{\Sigma}_1$, with $\lambda_1 > \lambda_2 > \ldots > \lambda_p$ the corresponding eigenvalues. Hence, let us consider several cases:

1. If $\boldsymbol{w} = \boldsymbol{v}_1$ is the dominant eigenvector of $\boldsymbol{\Sigma}_1$, then, from the properties of the Rayleigh quotient [40],

   $$\boldsymbol{w} = \text{argmax}_{\boldsymbol{z}} \boldsymbol{z}^\dagger \boldsymbol{\Sigma}_1 \boldsymbol{z}, \tag{B.12}$$

   and, therefore, $\sigma_1 > s_1$ and $\sigma_1 > \sigma_2$. It follows that

   $$\boldsymbol{z}^\dagger \Delta_{\boldsymbol{w}}^2 L \boldsymbol{z} > 0, \tag{B.13}$$

   and therefore $\boldsymbol{w}$ is a minimum of the L1 norm cost function.
2. Similarly, if $\boldsymbol{w} = \boldsymbol{v}_p$ is the least dominant eigenvector of $\boldsymbol{\Sigma}_1$ (the eigenvector associated with the smallest eigenvalue), then, from the Rayleigh quotient again [40], $\sigma_1 < s_1$ and $\sigma_1 < \sigma_2$. It follows that $\boldsymbol{z}^\dagger \Delta_{\boldsymbol{w}}^2 L \boldsymbol{z} > 0$ and $\boldsymbol{w}_p$ is still a minimum.
3. If $\boldsymbol{w} = \boldsymbol{v}_i$, $1 < i < p$, is any of the remaining eigenvectors, the sign of (B.11) when $\boldsymbol{z} = \boldsymbol{v}_1$ is different from that when $\boldsymbol{z} = \boldsymbol{v}_p$, and both $\boldsymbol{v}_1$ and $\boldsymbol{v}_p$ fulfill (B.4) because the eigenvectors are mutually orthogonal. That is, in the vicinity of $\boldsymbol{w} = \boldsymbol{v}_i$ the objective function increases in one direction and decreases in another. Therefore, $\boldsymbol{w}$ is a saddle point. Having said that, **if $\boldsymbol{w}$ is constrained to be orthogonal to $\boldsymbol{v}_1$ and $\boldsymbol{v}_p$, then it can be easily shown that $\boldsymbol{v}_2$ and $\boldsymbol{v}_{p-1}$are the new minima of the L1-cost and so on**. Hence, all the eigenvectors can be actually calculated by minimization techniques, constrained to be orthogonal to the previously calculated ones.

Finally note that Eq. (15) also has the solution $\sigma_1 = \sigma_2$. Let us briefly show that this solution corresponds to the absolute maximum of the L1-objective function. Let $\mathrm{b} = (\sigma_1, \sigma_2)^\dagger$, $\mathbf{1} = (1,1)^\dagger$ and $\boldsymbol{D} = \text{diag}(P(\mathcal{C}_1), P(\mathcal{C}_2))$. Define the weighted inner product $(\mathrm{b}, \mathbf{1})_{\boldsymbol{D}} = \mathrm{b}^\dagger \boldsymbol{D} \mathbf{1}$. Then, by the Cauchy-Schwarz inequality,

$$(\mathrm{b}, \mathbf{1})_{\boldsymbol{D}} \le \sqrt{(\mathrm{b}, \mathrm{b})_{\boldsymbol{D}}\, (\mathbf{1}, \mathbf{1})_{\boldsymbol{D}}} = \sqrt{\sum_{i=1,2} P(\mathcal{C}_i)\, \sigma_i^2} = 1.$$

where the final equality follows from (2). It is then easy to show the following inequality that restricts (A.3), i.e.,

$$\mathrm{E}\{|Y|\} = \sqrt{\frac{2}{\pi}} \sum_{k=1}^{2} P(\mathcal{C}_k)\, \sigma_k = \sqrt{\frac{2}{\pi}}\, (\mathrm{b}, \mathbf{1})_{\boldsymbol{D}} \le \sqrt{\frac{2}{\pi}},$$

with equality iff b is proportional to $\mathbf{1}$, implying $\sigma_1 = \sigma_2$. This completes the proof.

Appendix C. Upper bounds on approximation (18)

To discuss the approximation (18), where $\eta \in \mathbb{R}^+$ is a small positive constant, let us find an upper bound on the remainder. The exponential matrix is defined in classic textbooks as follows (e.g. see [40])

$$\boldsymbol{U} = \exp(-\eta \boldsymbol{S}) = \boldsymbol{I} + \sum_{k=1}^{\infty} \frac{(-\eta \boldsymbol{S})^k}{k!}$$

where $\boldsymbol{I}$ is the identity matrix and subscript n is omitted for simplicity. Consider approximating the exponential by $\hat{\boldsymbol{U}} = \boldsymbol{I} - \eta \boldsymbol{S}$. The approximation error is computed as

$$\boldsymbol{R} = \boldsymbol{U} - \hat{\boldsymbol{U}} = \sum_{k=2}^{\infty} \frac{(-\eta \boldsymbol{S})^k}{k!}.$$

By the triangle inequality property of matrix norms:

$$\begin{aligned} \|\mathbf{R}\| &\le \textstyle\sum_{k=2}^{\infty} \eta^k \frac{\|\boldsymbol{S}\|^k}{k!} = \eta^2 \|\mathbf{S}\|^2 \sum_{k=0}^{\infty} \frac{\varepsilon^k}{(k+2)!} \\ &< \eta^2 \|\mathbf{S}\|^2 \textstyle\sum_{k=0}^{\infty} \frac{\varepsilon^k}{k!} = \eta^2 \|\mathbf{S}\|^2 \exp(\varepsilon) \end{aligned}$$

where $\varepsilon = \eta \|\boldsymbol{S}\|$. Consequently, the approximation error norm is upper bounded by

$$\|\mathbf{R}\| \le \eta^2 \|\mathbf{S}\|^2 \exp(\eta \|\mathbf{S}\|).$$

which is dominated by a term of the order of $\eta^2 \|\boldsymbol{S}\|^2$ for small $\eta < \frac{1}{\|\boldsymbol{S}\|}$.

Appendix D. The gradient in the Stiefel manifold

For the reader's interest, let us summarize briefly the most remarkable properties of orthogonality constraints. Consider the set of all n-tuples of orthonormal vectors in $\mathbb{R}^p$. This set is known as the Stiefel manifold and is denoted by $V_{p,n}$ [26]. Alternatively, as an n-tuple $(\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n)$ of vectors in $\mathbb{R}^p$ can be regarded as a matrix $\boldsymbol{W} = [\boldsymbol{w}_1, \ldots, \boldsymbol{w}_n] \in \mathbb{R}^{p \times n}$, the manifold can be also expressed as $V_{p,n} = \{\boldsymbol{W} \in \mathbb{R}^{p \times n} : \boldsymbol{W}^\dagger \boldsymbol{W} = \boldsymbol{I}_n\}$, where $\boldsymbol{I}_n$ is the n-dimensional identity matrix.

Let $\boldsymbol{Q}(t)$ be a differentiable curve on $V_{p,n}$ with $\boldsymbol{Q}(0) = \boldsymbol{W} \in V_{p,n}$. The derivative $\dot{\boldsymbol{Q}}(0)$ can be regarded as the 'tangent vector' at $\boldsymbol{W}$ to the curve. The use of the term 'tangent' is justified because, intuitively, $\dot{\boldsymbol{Q}}(0)$ has the same direction as an infinitesimal displacement $d\boldsymbol{Q}(0)$ along the manifold. The tangent vectors calculated in this way, from each possible curve passing through $\boldsymbol{W}$, form a vector space called the *tangent space* at $\boldsymbol{W}$.

As $\boldsymbol{Q}(t)^\dagger \boldsymbol{Q}(t) = \boldsymbol{I}_p$ for all t, we readily find, after differentiating, that $\dot{\boldsymbol{Q}}(0)^\dagger \boldsymbol{Q}(0) + \boldsymbol{Q}(0)^\dagger \dot{\boldsymbol{Q}}(0) = \boldsymbol{0}$. From here, it follows that the tangent space at $\boldsymbol{W}$ is the set of matrices defined by

$$\mathcal{T}_{\boldsymbol{W}} V_{p,n} = \{\boldsymbol{S} \in \mathbb{R}^{p \times n} : \boldsymbol{S}^\dagger \boldsymbol{W} + \boldsymbol{W}^\dagger \boldsymbol{S} = \boldsymbol{0}\}.$$

Similarly, given any $p \times n$ matrix $\boldsymbol{Z}$, it can be also shown (see e.g. [26]) that

$$\pi_{\mathcal{T}_{\boldsymbol{W}}}(\boldsymbol{Z}) = \big(\boldsymbol{I}_p - \boldsymbol{W}\boldsymbol{W}^\dagger\big)\boldsymbol{Z} + \frac{1}{2}\boldsymbol{W}\big(\boldsymbol{W}^\dagger \boldsymbol{Z} - \boldsymbol{Z}^\dagger \boldsymbol{W}\big) \tag{D.1}$$

is the projection of $\boldsymbol{Z}$ onto $\mathcal{T}_{\boldsymbol{W}} V_{p,n}$. Now, consider the problem

$$\min_{\boldsymbol{W} \in \mathbb{R}^{p \times n}} J(\boldsymbol{W}) \text{ s.t. } \boldsymbol{W}^\dagger \boldsymbol{W} = \boldsymbol{I}_n.$$

Given the derivative $\partial J(\boldsymbol{W})$ of $J(\boldsymbol{W})$ at $\boldsymbol{W}$ in the Euclidean space, calculated element-wise, i.e., $(\partial J(\boldsymbol{W}))_{ij} = \frac{\partial J(\boldsymbol{W})}{\partial \boldsymbol{W}_{ij}}$, the gradient of $J(\boldsymbol{W})$ on the Stiefel manifold is obtained as the projection $\pi_{\mathcal{T}_{\boldsymbol{W}}}(\partial J(\boldsymbol{W}))$ given by Eq. (D.1). In the particular case of square matrices, $p = n$, $\boldsymbol{W}^\dagger \boldsymbol{W} = \boldsymbol{W}\boldsymbol{W}^\dagger = \boldsymbol{I}_p$ and

$$\pi_{\mathcal{T}_{\boldsymbol{W}}}(\partial J(\boldsymbol{W})) = \frac{1}{2}\big(\partial J(\boldsymbol{W}) - \boldsymbol{W}\partial J(\boldsymbol{W})^\dagger \boldsymbol{W}\big).$$

Observe finally that this formula is the same (up to the 1/2 constant) to the gradient that appears in Eq. (20). Therefore, the algorithm proposed in Section 5.2 computes the gradient-descent minimization of criterion J in the Stiefel manifold.

CRediT authorship contribution statement

José Luis Camargo: Conceptualization, Methodology, Software, Validation, Formal analysis. **Rubén Martín-Clemente:** Conceptualization, Methodology. **Susana Hornillo-Mellado:** Software, Validation, Formal analysis. **Vicente Zarzoso:** Conceptualization, Writing - review & editing.

References

[1] K. Fukunaga, W. Koontz, Application of the Karhunen-Loève expansion to feature selection and ordering, IEEE Trans. Comput. C-19 (4) (1970) 311–318.
[2] X. Huo, M. Elad, A.G. Flesia, R.R. Muise, S.R. Stanfill, J. Friedman, B. Popescu, J. Chen, A. Mahalanobis, D.L. Donoho, Optimal reduced-rank quadratic classifiers using the Fukunaga-Koontz transform with applications to automated target recognition, in: F.A. Sadjadi (Ed.), Automatic Target Recognition XIII, SPIE, 2003, pp. 59–72.
[3] X. Huo, A statistical analysis of Fkunaga–Koontz transform, IEEE Signal Process Lett 11 (2) (2004) 123–126.
[4] J. Peng, G. Seetharaman, W. Fan, S. Robila, A. Varde, Chernoff dimensionality reduction–where Fisher meets FKT, in: Proceedings of the 2011 SIAM International Conference on Data Mining, Society for Industrial and Applied Mathematics, 2011, pp. 271–282.
[5] R. Duin, M. Loog, Linear dimensionality reduction via a heteroscedastic extension of LDA: the Chernoff criterion, IEEE Trans Pattern Anal Mach Intell 26 (6) (2004) 732–739.
[6] T.M. Cover, J.A. Thomas, Elements of Information Theory, Wiley, 2005.
[7] A. Miranda, P. Whelan, Fukunaga-Koontz transform for small sample size problems, in: Proceedings of the IEE Irish Signals and Systems Conference (ISSC, Dublin, Ireland, 2005), 2005, pp. 1–6.
[8] S. Zhang, T. Sim, Discriminant subspace analysis: a Fukunaga-Koontz approach, IEEE Trans Pattern Anal Mach Intell 29 (10) (2007) 1732–1745.
[9] A. Bal, M.S. Alam, Automatic target tracking in forward-looking infrared video sequences using tuned basis functions, Opt. Eng. 55 (7) (2016) 073102.
[10] H. Binol, Improved Fukunaga–Koontz transform with compositional kernel combination for hyperspectral target detection, J. Indian Soc. Remote Sens. 46 (10) (2018) 1605–1615.
[11] F. Juefei-Xu, M. Savvides, Multi-class Fukunaga Koontz discriminant analysis for enhanced face recognition, Pattern Recognit 52 (2016) 186–205.
[12] R. Liu, E. Liu, J. Yang, Y. Zeng, F. Wang, Y. Cao, Automatically detect and track infrared small targets with kernel Fukunaga-Koontz transform and kalman prediction, Appl Opt 46 (31) (2007) 7780.
[13] S. Ochilov, M.S. Alam, A. Bal, Fukunaga-Koontz transform based dimensionality reduction for hyperspectral imagery, in: S.S. Shen, P.E. Lewis (Eds.), Algorithms and Technologies for Multispectral, Hyperspectral, and Ultraspectral Imagery XII, SPIE, 2006, pp. 1–8.
[14] Z.J. Koles, M.S. Lazar, S.Z. Zhou, Spatial patterns underlying population differences in the background EEG, Brain Topogr 2 (4) (1990) 275–284.
[15] W. Wu, Z. Chen, X. Gao, Y. Li, E.N. Brown, S. Gao, Probabilistic common spatial patterns for multichannel EEG analysis, IEEE Trans Pattern Anal Mach Intell 37 (3) (2015) 639–653.
[16] H. Binol, G. Bilgin, S. Dinc, A. Bal, Kernel Fukunaga–Koontz transform subspaces for classification of hyperspectral images with small sample sizes, IEEE Geosci. Remote Sens. Lett. 12 (6) (2015) 1287–1291.
[17] S. Hoell, P. Omenzetter, Fukunaga-Koontz feature transformation for statistical structural damage detection and hierarchical neuro-fuzzy damage localisation, J Sound Vib 400 (2017) 329–353.
[18] N. Kwak, Principal component analysis based on L1-norm maximization, IEEE Trans Pattern Anal Mach Intell 30 (9) (2008) 1672–1680.
[19] P.P. Markopoulos, G.N. Karystinos, D.A. Pados, Optimal algorithms for L1-subspace signal processing, IEEE Trans. Signal Process. 62 (19) (2014) 5046–5058.
[20] P.P. Markopoulos, S. Kundu, S. Chamadia, N. Tsagkarakis, D.A. Pados, Outlier-resistant data processing with L1-norm principal component analysis, in: Advances in Principal Component Analysis, Springer Singapore, 2017, pp. 121–135.
[21] R. Martín-Clemente, V. Zarzoso, On the link between L1-PCA and ICA, IEEE Trans Pattern Anal Mach Intell 39 (3) (2017) 515–528.
[22] R. Martín-Clemente, V. Zarzoso, LDA via L1-PCA of whitened data, IEEE Trans. Signal Process. 68 (2020) 225–240.
[23] I.T. Jolliffe, Principal component analysis, Springer, New York, NY, 2002.
[24] A. Kessy, A. Lewin, K. Strimmer, Optimal whitening and decorrelation, Am Stat 72 (4) (2018) 309–314.
[25] P.P. Markopoulos, S. Kundu, S. Chamadia, D.A. Pados, Efficient l1-norm principal-component analysis via bit flipping, IEEE Trans. Signal Process. 65 (16) (2017) 4252–4264.
[26] A. Edelman, T.A. Arias, S.T. Smith, The geometry of algorithms with orthogonality constraints, SIAM J. Matrix Anal. Appl. 20 (2) (1998) 303–353.
[27] J.-B. Hiriart-Urruty, C. Lemaréchal, Fundamentals of Convex Analysis, Springer Berlin Heidelberg, 2001.
[28] A.I. Fleishman, A method for simulating non-normal distributions, Psychometrika 43 (4) (1978) 521–532.
[29] C.D. Vale, V.A. Maurelli, Simulating multivariate nonnormal distributions, Psychometrika 48 (3) (1983) 465–471.
[30] J. Mathews, Numerical Methods Using MATLAB, Prentice Hall, Upper Saddle River, N.J, 1999.
[31] P.J. Rousseeuw, K.V. Driessen, A fast algorithm for the minimum covariance determinant estimator, Technometrics 41 (3) (1999) 212–223.
[32] F. Lotte, L. Bougrain, A. Cichocki, M. Clerc, M. Congedo, A. Rakotomamonjy, F. Yger, A review of classification algorithms for EEG-based brain–computer interfaces: a 10 year update, J Neural Eng 15 (3) (2018) 031005.
[33] R. Martín-Clemente, J. Olias, D. Thiyam, A. Cichocki, S. Cruces, Information theoretic approaches for motor-imagery BCI systems: review and experimental comparison, Entropy 20 (1) (2018) 7.
[34] B. Blankertz, C. Vidaurre, M. Tangermann, K.-R. Müller, C. Brunner, R. Leeb, G. Müller-Putz, A. Schlögl, G. Pfurtscheller, S. Waldert, C. Mehring, A. Aertsen, G.S. Niels Birbaumer K. J. Miller BCI Competition IV dataset, 2008, (http://www.bbci.de/competition/iv/). accessed April 2020.
[35] R. Leeb, F. Lee, C. Keinrath, R. Scherer, H. Bischof, G. Pfurtscheller, Brain–computer communication: motivation, aim, and impact of exploring a virtual apartment, IEEE Trans. Neural Syst. Rehabil. Eng. 15 (4) (2007) 473–482.
[36] M. Tangermann, K.-R. Müller, A. Aertsen, N. Birbaumer, C. Braun, C. Brunner, R. Leeb, C. Mehring, K.J. Miller, G.R. Müller-Putz, G. Nolte, G. Pfurtscheller, H. Preissl, G. Schalk, A. Schlögl, C. Vidaurre, S. Waldert, B. Blankertz, Review of the BCI competition IV, Front Neurosci 6 (2012).
[37] J. Olias, R. Martín-Clemente, M.A. Sarmiento-Vega, S. Cruces, EEG Signal processing in MI-BCI applications with improved covariance matrix estimators, IEEE Trans. Neural Syst. Rehabil. Eng. 27 (5) (2019) 895–904.
[38] I.S. Gradshteyn, I. Ryzhik, Table of Integrals, Series and Products, Academic, Oxford, 2007.
[39] E.K.P. Chong, S.H. Zak, An introduction to optimization, John Wiley & Sons, 2013.
[40] G. Golub, C. Van Loan, Matrix Computations, Johns Hopkins University Press, Baltimore, 1996.