Neurocomputing 123 (2014) 49–59

Contents lists available at ScienceDirect

Neurocomputing

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

Generative embeddings based on Rician mixtures for kernel-based classification of magnetic resonance images

Anna C. Carli [a,*], Mário A.T. Figueiredo [b], Manuele Bicego [a], Vittorio Murino [a,c]

[a] Dipartimento di Informatica, Università di Verona, Verona, Italy
[b] Instituto de Telecomunicações, Instituto Superior Técnico, Lisboa, Portugal
[c] Istituto Italiano di Tecnologia (IIT), Genova, Italy

ARTICLE INFO

Article history:
Received 26 April 2012
Received in revised form
21 February 2013
Accepted 23 February 2013
Available online 11 April 2013

Keywords:
Rician mixture
EM algorithm
Generative embedding
Discriminative learning
Information theory
Boosting

ABSTRACT

Classical approaches to classifier learning for structured objects (such as images or sequences) are based on probabilistic generative models. On the other hand, state-of-the-art classifiers for vectorial data are learned discriminatively. In recent years, these two dual paradigms have been combined via the use of generative embeddings (of which the Fisher kernel is arguably the best known example); these embeddings are mappings from the object space into a fixed dimensional score space, induced by a generative model learned from data, on which a (maybe kernel-based) discriminative approach can then be used.

This paper proposes a new semi-parametric approach to build generative embeddings for classification of magnetic resonance images (MRI). Based on the fact that MRI data is well described by Rice distributions, we propose to use Rician mixtures as the underlying generative model, based on which several different generative embeddings are built. These embeddings yield vectorial representations on which kernel-based support vector machines (SVM) can be trained for classification. Concerning the choice of kernel, we adopt the recently proposed nonextensive information theoretic kernels.

The methodology proposed was tested on a challenging classification task, which consists in classifying MRI images as belonging to schizophrenic or non-schizophrenic human subjects. The classification is based on a set of regions of interest (ROIs) in each image, with the classifiers corresponding to each ROI being combined via AdaBoost. The experimental results show that the proposed methodology outperforms the previous state-of-the-art methods on the same dataset.

1. Introduction

Classical approaches to learning classifiers follow one of two paradigms: generative and discriminative [1,2]. Generative approaches are based on probabilistic class models and *a priori* class probabilities, learnt from training data and combined via Bayes law to yield posterior probability estimates. Discriminative methods learn class boundaries or posterior class probabilities directly from data, without using generative class models.

In the past decade, several hybrid generative–discriminative approaches have been proposed, aiming at taking advantage of the best of both paradigms [3,4]. In this context, the so-called generative score space methods (or generative embeddings) have sparked significant interest. The idea is to exploit a generative model to map the objects to be classified into a space where discriminative (e.g., kernel-based) techniques can be used. This scheme is particularly suitable to deal with non-vectorial data (strings, trees, images), since it maps objects (maybe of different dimensions) into a fixed dimension space.

Prior knowledge about the underlying data generation mechanism can be embedded in the kernel in different ways. The Fisher kernel [3], arguably the seminal work on generative embeddings, considers a fixed probability distribution and obtains the features of a given object as the derivatives of the log-likelihood with respect to the model parameters, computed at that object. Marginalization kernels assume that there is some hidden model that governs the data generation and marginalize with respect to this model [5–7]. Kernels can also be devised between probability measures [8–11], by mapping data to points into a probability space; more generally, kernels may be defined between unnormalized measures [12–14]. Some of these kernels use classical information-theoretic quantities, e.g., the Jensen–Shannon divergence. More recently, grounded on nonextensive generalizations of Shannon's information theory [15], a new family

* Corresponding author. Present address: Ministero dello Sviluppo Economico, Dipartimento per le Comunicazioni, Viale America, 201, 00144 Roma, Italy. Tel.: +39 06 54442489.
E-mail addresses: annacaterina.carli@mise.gov.it, annacaterina.carli@gmail.com (A.C. Carli).

0925-2312/$ - see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.neucom.2013.02.037

of nonextensive information-theoretic kernels was proposed [14]. Those kernels are based on the Jensen–Tsallis q-difference, a nonextensive generalization of the Jensen–Shannon divergence obtained through the new concept of q-convexity and a related q-Jensen inequality.

In this paper, we exploit generative embeddings to tackle a challenging classification task: based on a set of regions of interest (ROIs) of a magnetic resonance image (MRI), classify the patient as suffering, or not, from schizophrenia [16].

We build on the well-known fact that MRI magnitude data (in homogenous regions) follows a Rician distribution. Statistical characteristics of MRI magnitude and phase values have been studied in the literature and analytical expressions have been derived [17,18], based on the noise response of the in-phase and quadrature demodulators, previously analyzed in telecommunications [19,20]. If the acquired real (in-phase) and imaginary (quadrature) images are corrupted by zero mean Gaussian stationary noise, the probability density function of the magnitude follows a Rician distribution. Other less accurate models have been shown to yield underestimation of the true noise power [17]. If homogenous MRI data follows a Rician distribution, an image composed of several regions naturally follows a mixture of Rician distributions [21–23], and that is precisely the model that we adopt in this paper. Based on this model, we propose several generative embeddings, aiming at fully exploiting this known statistical model of MRI data.

The proposed generative mappings referred in the previous paragraph allow learning kernel-based classifiers. In this paper, we propose learning a support vector machine (SVM) classifier for each ROI. We adopt the nonextensive information-theoretic kernels, recently proposed in [14], which are a good fit to the probabilistic nature to the generative embeddings. Finally, an optimal combination of these SVM classifiers is sought via the AdaBoost algorithm [24]. The experimental results show that the proposed methodology outperforms the previous state-of-the-art on the same dataset.

The paper is organized as follows. Section 2 addresses the problem of estimating Rician mixtures via the expectation–maximization (EM) algorithm. In Section 3, we propose several generative embeddings using Rician mixture models. Section 4 briefly reviews the information theoretic kernels proposed by [14], while Section 5 describes SVM combination by boosting. Finally, Section 6 reports experimental results on the MRI categorization problem. Finally, we should mention that a preliminary version of the work reported in this paper appeared in our earlier conference publication [25].

2. Rician mixture fitting via the EM algorithm

This section presents the derivation of the EM algorithm for estimating the parameters of a Rician mixture; the main novelty in this derivation is that it yields closed-form parameter update expressions [25], whereas in previous work the M-step is implemented via numerical optimization (for example, a quasi-Newton method [21]). Related work can be found in [26], where the problem of estimating a mixture of one Rician and one uniform density is addressed; also there, the M-step is solved numerically, via a Newton–Raphson algorithm. Finally, after our earlier work that contains a similar derivation was published in [25] and this paper was submitted for publication, a related algorithm appeared in [27].

A Rician probability density function [19] has the form

$$f_R(y;\nu,\sigma)=\frac{y}{\sigma^2}e^{-(y^2+\nu^2)/2\sigma^2}I_0\left(\frac{y\nu}{\sigma^2}\right), \tag{1}$$

for $y>0$, and zero for $y\le 0$, where ν is the magnitude parameter, σ is the noise parameter, and $I_0(z)$ denotes the 0-th order modified Bessel function of the first kind [28].

A mixture of g Rician densities has the form

$$f(y;\boldsymbol{\Psi})=\sum_{i=1}^{g}\pi_i f_R(y;\nu_i,\sigma_i^2), \tag{2}$$

where $\pi_i\ge 0$, for $i=1,\dots,g$, are quantities that sum to one (the so-called mixing weights), $\boldsymbol{\Psi}=(\pi_1,\dots,\pi_{g-1},\boldsymbol{\theta}_1,\dots,\boldsymbol{\theta}_g)$ is the vector of all the parameters of the mixture, and $\boldsymbol{\theta}_i=(\nu_i,\sigma_i^2)$ is the pair of parameters of component i.

Let $Y=\{y_1,\dots,y_n\}$ be a random sample of size n, assumed to have been generated independently by a mixture of the form (2) and consider the goal of obtaining a maximum likelihood estimate (MLE) of $\boldsymbol{\Psi}$, that is, $\widehat{\boldsymbol{\Psi}}=\arg\max_{\Psi}L(\boldsymbol{\Psi})$, where

$$L(\boldsymbol{\Psi},Y)=\sum_{j=1}^{n}\log f(y_j;\boldsymbol{\Psi})=\sum_{j=1}^{n}\log\sum_{i=1}^{g}\pi_i f_R(y_j;\nu_i,\sigma_i^2). \tag{3}$$

The expectation–maximization (EM) algorithm is the most common approach for computing the MLE of the parameters of a finite mixture [29–33]. As is common in EM, let $\mathbf{z}_j\in\{0,1\}^g$ be a g-dimensional hidden/missing binary label vector associated to observation y_j, such that $z_{ji}=1$ if and only if y_j was generated by the i-th mixture component. The so-called complete data is $\{(y_1,\mathbf{z}_1),\dots,(y_n,\mathbf{z}_n)\}$ and the corresponding complete loglikelihood for $\boldsymbol{\Psi}$, $\log L_c(\boldsymbol{\Psi})$, is given by

$$L_c(\boldsymbol{\Psi},Y,Z)=\sum_{j=1}^{n}\sum_{i=1}^{g}z_{ji}\left\{\log\pi_i+\log f_R(y_j;\boldsymbol{\theta}_i)\right\} \tag{4}$$

where $Z=\{\mathbf{z}_1,\dots,\mathbf{z}_n\}$.

The EM algorithm proceeds iteratively in two steps. The E-step computes the conditional expectation (with respect to the missing labels Z) of the complete loglikelihood given the observed data Y and the current parameter estimate $\widehat{\boldsymbol{\Psi}}^{(k)}$,

$$Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)}):=\mathbb{E}_Z\left[L_c(\Psi,Y,Z)|Y,\widehat{\boldsymbol{\Psi}}^{(k)}\right]. \tag{5}$$

Since $L_c(\boldsymbol{\Psi},Y,Z)$ is linear in the missing data z_{ji} (see (4)), this reduces to computing the conditional expectation of the z_{ji} and plugging these into the complete loglikelihood. Each of these conditional expectations (denoted w_{ji}) is equal to the posterior probability that the j-th sample was generated by the i-th component of the mixture,

$$w_{ji}=\frac{\pi_i f_R(y_j;\boldsymbol{\theta}_i^{(k)})}{\sum_{h=1}^{g}\pi_h^{(k)}f_R(y_j;\boldsymbol{\theta}_h^{(k)})}, \tag{6}$$

for $i=1,\dots,g$ and $j=1,\dots,n$. It follows that the conditional expectation of the complete loglikelihood (5) becomes

$$Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)})=\sum_{i=1}^{g}\sum_{j=1}^{n}w_{ji}\left\{\log\pi_i+\log f_R(y_j;\boldsymbol{\theta}_i)\right\}. \tag{7}$$

The M-step obtains an updated parameter estimate $\boldsymbol{\Psi}^{(k+1)}$ by maximizing $Q(\boldsymbol{\Psi};\boldsymbol{\Psi}^{(k)})$ with respect to $\boldsymbol{\Psi}$. The updated estimates of the mixing weights $\pi_i^{(k+1)}$ are well-known to be

$$\pi_i^{(k+1)}=\frac{1}{n}\sum_{j=1}^{n}w_{ji}. \tag{8}$$

2.1. Updating the parameters of the Rician components

Updating the estimate of $\boldsymbol{\theta}_i=(\nu_i,\sigma_i^2)$ requires solving

$$\sum_{i=1}^{g}\sum_{j=1}^{n}w_{ji}\nabla_\theta\log f_R(y_j;\boldsymbol{\theta}_i)=0, \tag{9}$$

where ∇_θ denotes the gradient with respect to $\boldsymbol{\theta}$. In the following proposition (proved in the Appendix), we provide a closed-form solution of (9) for the Rician mixture.

Proposition 2.1. *The updated estimate* $\widehat{\boldsymbol{\theta}}_i^{(k+1)} = (\widehat{v}_i^{(k+1)}, (\widehat{\sigma}_i^2)^{(k+1)})$, *that is, the solution of* (9), *is*

$$\widehat{v}_i^{(k+1)} = \frac{\sum_{j=1}^n w_{ji}\, y_j \phi\left(\frac{y_j v_i^{(k)}}{\sigma_i^{2^{(k)}}}\right)}{\sum_{j=1}^n w_{ji}} \tag{10}$$

and

$$(\widehat{\sigma}_i^2)^{(k+1)} = \frac{\sum_{j=1}^n w_{ji}\left(y_j^2 + v_i^{(k+1)^2} - 2 y_j v_i^{(k+1)} \phi\left(\frac{y_j v_i^{(k)}}{\sigma_i^{2^{(k)}}}\right)\right)}{2\sum_{j=1}^n w_{ji}} \tag{11}$$

where $\phi(u) = I_1(u)/I_0(u)$.

Finally, we refer that in our experiments, we use the classical random initialization of EM; since we are dealing with univariate mixtures with a few components, initialization is not a critical issue.

3. Generative embeddings based on Rician mixtures

We now introduce several generative embeddings for MR images, based on Rician mixture models. Let $\{X_1, \ldots, X_S\}$ be a set of images or ROIs (each belonging to one or R classes, $c_s \in \{1, \ldots, R\}$), where each image $X_s = \{y_1^s, \ldots, y_{N_s}^s\}$ is simply modeled as a bag of N_s strictly positive pixels $y_j^s \in \mathbb{R}_{++}$, for $j = 1, \ldots, N_s$. Let $\mathcal{X}$ denote the input domain, that is, a set to which all these images belong. We map objects in $\mathcal{X}$ into a finite-dimensional Hilbert space $\mathcal{H}$ (the so-called *generative embedding space*) using the Rician mixture generative model; formally,

$$\begin{aligned} e: \quad & \mathcal{X} \longrightarrow \mathcal{H} \\ & X_s \longmapsto \mathbf{e}(X_s; \boldsymbol{\Psi}) \in \mathcal{H}. \end{aligned} \tag{12}$$

The embedding $\mathbf{e}(X_s; \boldsymbol{\Psi})$ depends on the parameters $\boldsymbol{\Psi}$ of a K-components Rician mixture, as explained next.

Based on a K-components Rician mixture with parameters $\boldsymbol{\Psi}$, the posterior probability that y_j^s (the j-th pixel of the s-th image) belongs to the i-th component of the mixture is (see (6))

$$w_i(y_j^s; \boldsymbol{\Psi}) = \pi_i f(y_j^s; \boldsymbol{\theta}_i) \left(\sum_{k=1}^K \pi_k f(y_j^s; \boldsymbol{\theta}_k) \right)^{-1} \tag{13}$$

Based on (13), six generative embeddings will now be defined.

Definition 3.1. With a single Rician mixture $\boldsymbol{\Psi}$ estimated for the S images, the embedding of an image $X = \{y_1, \ldots, y_N\}$ is a K-dimensional vector given by

$$\widetilde{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}) = \frac{1}{N}\left[\sum_{j=1}^N w_1(y_j; \boldsymbol{\Psi}), \ldots, \sum_{j=1}^N w_K(y_j; \boldsymbol{\Psi})\right]^T. \tag{14}$$

Notice that this embedding always yields a vector of non-negative values that sum to one, thus it can be interpreted as a discrete probability measure.

Definition 3.2. With R Rician mixtures (one per class) $\{\boldsymbol{\Psi}_1, \ldots, \boldsymbol{\Psi}_R\}$, each with K components, the embedding of an image $X = \{y_1, \ldots, y_N\}$ is a (KR)-dimensional vector:

$$\widetilde{\mathbf{e}}(X; \boldsymbol{\Psi}_1, \ldots, \boldsymbol{\Psi}_R) = \frac{1}{N}\left[\left(\widetilde{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}_1)\right)^T, \ldots, \left(\widetilde{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}_R)\right)^T\right]^T. \tag{15}$$

Definition 3.3. We will also consider the two following K-dimensional embeddings, defined for an arbitrary image $X = \{y_1, \ldots, y_N\}$ as

$$\overline{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}) = \frac{1}{N}\sum_{j=1}^N \left[\pi_1 f(y_j; \boldsymbol{\theta}_1), \ldots, \pi_K f(y_j; \boldsymbol{\theta}_K)\right]^T$$

and

$$\widehat{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}) = \frac{1}{N}\sum_{j=1}^N \left[f(y_j; \boldsymbol{\theta}_1), \ldots, f(y_j; \boldsymbol{\theta}_K)\right]^T,$$

as well as their (KR)-dimensional generalizations to the case in which a Rician mixture is estimated for each of the R classes,

$$\overline{\mathbf{e}}(X; \boldsymbol{\Psi}_1, \ldots, \boldsymbol{\Psi}_R) = \left[\left(\overline{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}_1)\right)^T, \ldots, \left(\overline{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}_R)\right)^T\right]^T$$

and

$$\widehat{\mathbf{e}}(X; \boldsymbol{\Psi}_1, \ldots, \boldsymbol{\Psi}_R) = \left[\left(\widehat{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}_1)\right)^T, \ldots, \left(\widehat{\mathbf{e}}^{\text{single}}(X; \boldsymbol{\Psi}_R)\right)^T\right]^T.$$

Notice that $\widetilde{\mathbf{e}}$, $\overline{\mathbf{e}}^{\text{single}}$, $\widehat{\mathbf{e}}^{\text{single}}$, $\overline{\mathbf{e}}$, and $\widehat{\mathbf{e}}$ yield vectors of non-negative values, thus interpretable as discrete unnormalized measures.

4. Nonextensive information theoretic kernels

This section briefly reviews the nonextensive information theoretic kernels proposed in [14] and introduces relevant notation. These kernels on measures are based on the Jensen–Tsallis q-difference, a nonextensive generalization of the Jensen–Shannon divergence obtained through the concept of q-convexity and a related q-Jensen inequality [14]. The motivation for the use of these information-theoretic kernels is the following: since the six proposed embeddings can be naturally interpreted as discrete measures (one normalized and five unnormalized), kernels between (possibly unnormalized) measures are a natural choice to use these embeddings in kernel-based learning algorithms.

4.1. Suyari's entropies

Both the Shannon–Boltzmann–Gibbs (SBG) and the Tsallis entropies are particular cases of functions $S_{q,\phi}$ following Suyari's axioms [34]. Let Δ^{n-1} be the standard probability simplex and $q \geq 0$ be a fixed scalar (the *entropic index*). The function $S_{q,\phi}: \Delta^{n-1} \to \mathbb{R}$ has the form

$$S_{q,\phi}(p_1, \ldots, p_n) = \begin{cases} \dfrac{k}{\phi(q)}\left(1 - \sum_{i=1}^n p_i^q\right) & \text{if } q \neq 1 \\ -k \sum_{i=1}^n p_i \ln p_i & \text{if } q = 1, \end{cases} \tag{16}$$

where $\phi: \mathbb{R}_+ \to \mathbb{R}$ is a continuous function with properties stated in [34], and $k > 0$ an arbitrary constant, henceforth set to $k = 1$. As is clear in (16), for $q = 1$, we recover the SBG entropy, while setting $\phi(q) = q - 1$ yields the Tsallis entropy

$$S_q(p_1, \ldots, p_n) = \frac{1}{q-1}\left(1 - \sum_{i=1}^n p_i^q\right) = -\sum_{i=1}^n p_i^q \ln_q p_i,$$

where $\ln_q(x) = (x^{1-q} - 1)/(1-q)$ is the q-logarithm function.

4.2. Jensen–Shannon (JS) divergence

Consider two measure spaces $(\mathcal{X}, \mathcal{M}, \nu)$, and $(\mathcal{T}, \mathcal{J}, \tau)$, where the second is used to index the first. Let H denote the SBG entropy, and consider the random variables $T \in \mathcal{T}$ and $X \in \mathcal{X}$, with densities $\pi(t)$

and $p(x) \triangleq \int_{\mathcal{T}} p(x|t)\pi(t)$. The Jensen divergence [14] is defined as

$$J^{\pi}(p) \triangleq J_H^{\pi}(p) = H(\mathbb{E}[p]) - \mathbb{E}[H(p)]. \tag{17}$$

When $\mathcal{X}$ and $\mathcal{T}$ are finite with $|\mathcal{T}| = m$, $J_H^{\pi}(p_1, \ldots, p_m)$ is called the *Jensen–Shannon (JS) divergence* of $p_1, \ldots, p_m$, with weights $\pi_1, \ldots, \pi_m$ [35,36]. In particular, if $|\mathcal{T}| = 2$ and $\pi = (1/2, 1/2)$, p may be seen as a random distribution whose value on $\{p_1, p_2\}$ is chosen tossing a fair coin. In this case, $J^{(1/2,1/2)} = JS(p_1, p_2)$, where

$$JS(p_1, p_2) \triangleq H\left(\frac{p_1 + p_2}{2}\right) - \frac{H(p_1) + H(p_2)}{2},$$

which will be used in Section 4.4 to define JS kernels.

4.3. Jensen–Tsallis (JT) q-differences

Tsallis' entropy can be written as $S_q(X) = -\mathbb{E}_q[\ln_q p(X)]$, where $\mathbb{E}_q$ denotes the *unnormalized q-expectation*, which, for a discrete random variable $X \in \mathcal{X}$ with probability mass function $p : \mathcal{X} \to \mathbb{R}$, is defined as

$$\mathbb{E}_q[X] \triangleq \sum_{x \in \mathcal{X}} x p(x)^q;$$

(of course, $\mathbb{E}_1[X]$ is the standard expectation).

As in Section 4.2, consider two random variables $T \in \mathcal{T}$ and $X \in \mathcal{X}$, with densities $\pi(t)$ and $p(x) \triangleq \int_{\mathcal{T}} p(x|t)\pi(t)$. The Jensen q-difference is the nonextensive analogue of (17) [14],

$$T_q^{\pi}(p) = S_q(\mathbb{E}[p]) - \mathbb{E}_q[S_q(p)].$$

If $\mathcal{X}$ and $\mathcal{T}$ are finite with $|\mathcal{T}| = m$, $T_q^{\pi}(p_1, \ldots, p_m)$ is called the *Jensen–Tsallis (JT) q-difference* of $p_1, \ldots, p_m$, with weights $\pi_1, \ldots, \pi_m$. In particular, if $|\mathcal{T}| = 2$ and $\pi = (1/2, 1/2)$, define $T_q = T_q^{1/2,1/2}$:

$$T_q(p_1, p_2) = S_q\left(\frac{p_1 + p_2}{2}\right) - \frac{S_q(p_1) + S_q(p_2)}{2},$$

which will be used in Section 4.4 to define JT kernels. Naturally, T_1 coincides with the JS divergence.

4.4. Jensen–Shannon and Tsallis kernels

The JS and JT differences underlie the kernels proposed in [14], which apply to normalized or unnormalized measures.

Definition 4.1 (*Weighted Jensen–Tsallis kernels*). Let μ_1 and μ_2 be two (not necessarily probability) measures; the kernel k_q is defined as

$$\widetilde{k}_q(\mu_1, \mu_2) \triangleq (S_q(\pi) - T_q^{\pi}(p_1, p_2))(\omega_1 + \omega_2)^q$$

where $p_1 = \mu_1/\omega_1$ and $p_2 = \mu_2/\omega_2$ are the normalized counterparts of μ_1 and μ_2 (which have total masses ω_1 and ω_2), and $\pi = (\omega_1 + \omega_2)^{-1}[\omega_1, \omega_2]$. The kernel k_q is defined as

$$k_q(\mu_1, \mu_2) \triangleq S_q(\pi) - T_q^{\pi}(p_1, p_2).$$

Notice that if $\omega_1 = \omega_2$, $\widetilde{k}_q$ and k_q coincide up to a scale factor. For $q = 1$, k_q is the so-called Jensen–Shannon kernel, $k_{JS}(p_1, p_2) = \ln 2 - JS(p_1, p_2)$.

The following proposition (proved in [14]) characterizes these kernels in terms of positive definiteness, a crucial aspect for their use in support vector machines (SVM) [14].

Proposition 4.1. *The kernel $\widetilde{k}_q$ is positive definite (pd), for $q \in [0, 2]$. The kernel k_q is pd, for $q \in [0, 1]$. The kernel k_{JS} is pd.*

In our approach, the information theoretic kernels are applied to the Rician generative embeddings $\mathbf{e}(X; \mathbf{\Psi})$ proposed in Section 3. This corresponds to an implicit mapping from the generative embedding space $\mathcal{H}$ to a so-called feature space $\mathcal{F}$, where the kernel corresponds to an inner product [37,38], that is,

$$\begin{aligned} \boldsymbol{\phi} &: \mathcal{H} \longrightarrow \mathcal{F} \\ &\mathbf{e}(X; \mathbf{\Psi}) \longmapsto \phi(\mathbf{e}(X; \mathbf{\Psi})) \in \mathcal{F} \end{aligned}$$

where $k(X_i, X_j) = \langle \phi(\mathbf{e}(X_i; \mathbf{\Psi})), \phi(\mathbf{e}(X_j; \mathbf{\Psi})) \rangle_{\mathcal{F}}$.

5. Combining SVM classifiers via boosting

The final building block of our approach to MR image classification is a way to combine the classifiers working on each of the several regions of interest (ROI). For that end, we adopt the AdaBoost algorithm [24], which we now briefly review. In the description of AdaBoost in Algorithm 5.1, each (weak) classifier $G_m(x)$, $m = 1, \ldots, M$, corresponds to one of the M regions.

Algorithm 5.1. AdaBoost [24]

1. Initialize weights $p_i = 1/S$, $i = 1, \ldots, S$.
2. For $m=1$ to M:
   (a) Learn classifier $G_m(x)$ with current weights.
   (b) Compute weighted error rate:
   $$\mathrm{err}_m = \frac{\sum_{i=1}^{S} p_i \mathbb{1}_{(y_i \neq G_m(x_i))}}{\sum_{i=1}^{S} p_i}.$$
   (c) Compute $\gamma_m = \log(1 - \mathrm{err}_m) - \log(\mathrm{err}_m)$.
   (d) $p_i \leftarrow p_i \cdot \exp(\gamma_m \mathbb{1}_{(y_i \neq G_m(x_i))})$, $i = 1, \ldots, S$.
3. Output $G(x) = \mathrm{sign}\left[\sum_{m=1}^{M} \gamma_m G_m(x)\right]$.

In the description of Algorithm 5.1, $\mathbb{1}_A$ is the usual indicator function: $\mathbb{1}_A = 1$, if A is true, and zero otherwise. Each boosting step requires learning a classifier by minimizing a weighted criterion, with weights $p_1, \ldots, p_S$ corresponding to each training observation (y_s, X_s), $s = 1, \ldots, S$. In our case, the classifier G_m is a weighted version of the SVM classifier corresponding to the m-th ROI, i.e., the SVM classifier whose kernel function is built on the Rician mixture estimated for that ROI. To take into account these weights, the optimization problem solved by the SVM learning algorithm requires a modification: the penalty on the slack variable ξ_i corresponding to the example X_i is set to be proportional to the weight p_i. The corresponding modified 1-norm SVM optimization problem (see [37,38] for details) is

$$\begin{aligned} \min_{\xi, \boldsymbol{\beta}, \beta_0} \quad & \langle \boldsymbol{\beta}, \boldsymbol{\beta} \rangle + C \sum_{i=1}^{S} p_i \, \xi_i \\ \text{s.t.} \quad & y_i(\langle \boldsymbol{\beta}, \boldsymbol{\phi}(X_i) \rangle + \beta_0) \geq 1 - \xi_i, \quad i = 1, \ldots, S \\ & \xi_i \geq 0, \quad i = 1, \ldots, S. \end{aligned} \tag{18}$$

The Lagrangian for problem (18) is

$$\begin{aligned} L_p(\boldsymbol{\beta}, \beta_0, \boldsymbol{\xi}, \boldsymbol{\alpha}, \boldsymbol{\mu}) &= \frac{1}{2}\|\boldsymbol{\beta}\|^2 + C \sum_{i=1}^{S} p_i \xi_i \\ &- \sum_{i=1}^{S} \alpha_i [y_i(\langle \boldsymbol{\phi}(X_i), \boldsymbol{\beta} \rangle + \beta_0) - (1 - \xi_i)] - \sum_{i=1}^{S} \mu_i \xi_i \end{aligned} \tag{19}$$

with $\alpha_i \geq 0$ and $\mu_i \geq 0$. By minimizing L_p with respect to $\boldsymbol{\beta}$, β_0, ξ_i and μ_i, $i = 1, \ldots, S$, the Lagrange dual problem results

$$\begin{aligned} \max_{\boldsymbol{\alpha}} \quad & \sum_{i=1}^{S} \alpha_i - \frac{1}{2} \sum_{i,j=1}^{S} \alpha_i \alpha_j y_i y_j k(X_i, X_j) \\ \text{s.t.} \quad & 0 \leq \alpha_i \leq p_i C \\ & \sum_{i=1}^{S} \alpha_i y_i = 0. \end{aligned} \tag{20}$$

Notice that each α_i is constrained to be less or equal to $p_i C$ (rather than simply C in the unweighted SVM) while the objective function in (20) remains unchanged [37,38]. Consequently, if p_i is small, so is α_i, thus contributing very weakly to the definition of the optimal hyperplane, which is still given by

$$f(X, \boldsymbol{\alpha}^*, \beta_0^*) = \sum_{i=1}^{S} y_i \alpha_i^* k(X_i, X) + \beta_0^*. \tag{21}$$

6. Experiments

We begin this section by summarizing the proposed approach. The training data consists of set of images, each labeled as belonging to a schizophrenic or non-schizophrenic subject, and containing a set of M regions of interest (ROI). For each ROI in the training set, either a single Rician mixture or two Rician mixtures (one per class) are estimated and used to embed the data on a Hilbert space, as described in Section 3. On the Hilbert space for each ROI, one of the information theoretic kernels described in Section 4 is used. Finally, a set of M (one per ROI) SVM classifiers is obtained by the AdaBoost algorithm described in Section 5; the final classifier is the one resulting at the last step of Algorithm 5.1.

The baselines against which we compare the proposed approach are SVM classifiers with linear kernels (LK) and Gaussian radial basis function kernels (GRBFK), on the same generative embeddings. SVM training is carried out using the LIBSVM package (http://www.csie.ntu.edu.tw/~cjlin/libsvm). The underlying Rician mixtures are estimated using the EM algorithm described in Section 2, with K (the number of components) selected by the criterion proposed in [39], which leads to numbers in the [4,6] range. Fig. 1 shows examples of fitted Rician mixtures for different ROIs, in the case

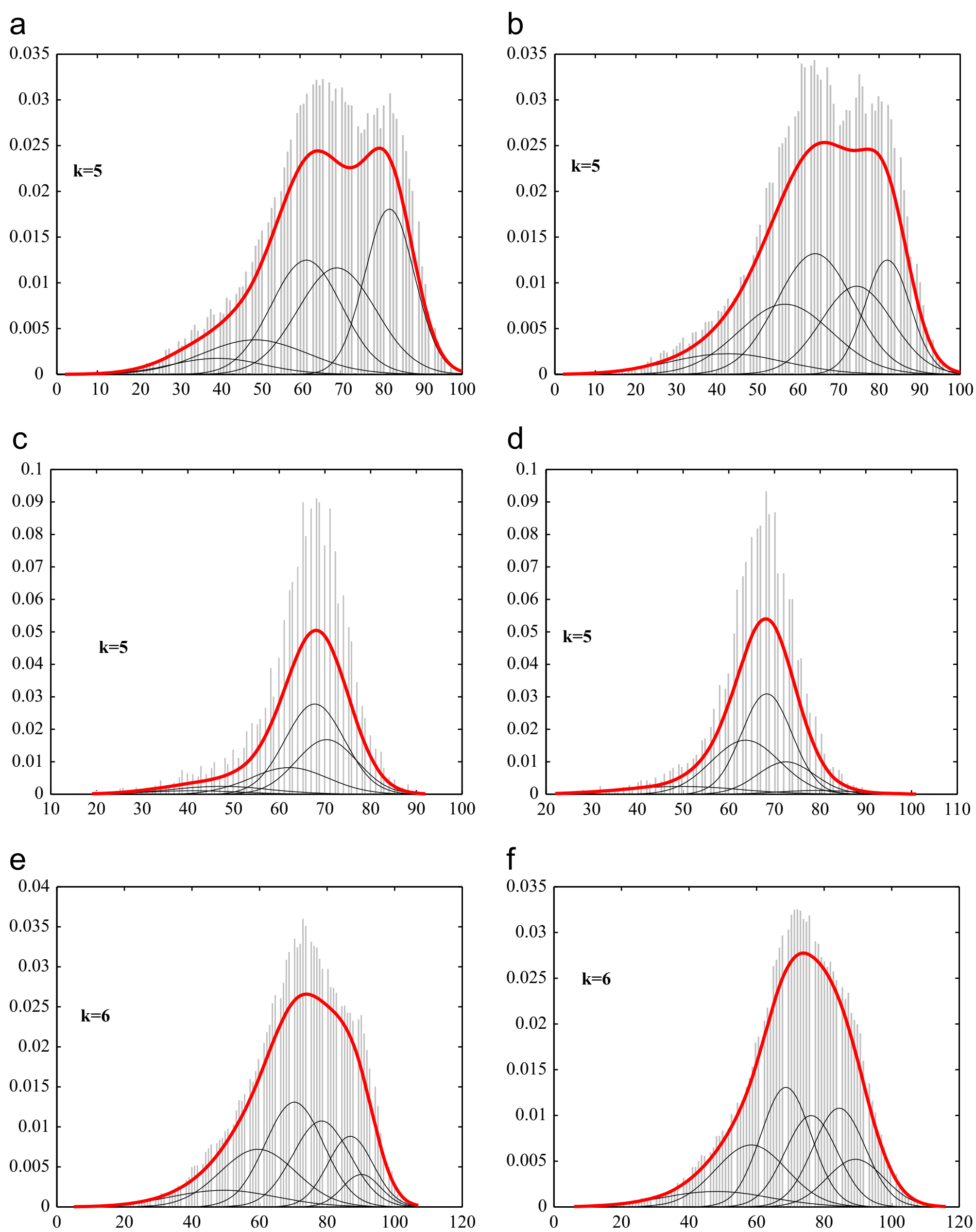

Fig. 1. Rician mixture fitting (one per class—schizophrenic on the left and non-schizophrenic on the right), for ROI 4 ((a), (b)), ROI 10 ((c), (d)), and ROI 12 ((e), (f)); k denotes the number of components in the mixture.

Table 1
Mean accuracy for the best values of q and C for the SVM classifiers learnt on ROI 2, 4, 6, 8, 12, 14 respectively, using one Rician mixture per class with $K=4,5,6$ components and embeddings $\widetilde{\mathbf{e}}$, $\overline{\mathbf{e}}$ and $\widehat{\mathbf{e}}$.

	Embedding	$\widetilde{\mathbf{e}}$			$\overline{\mathbf{e}}$			$\widehat{\mathbf{e}}$		
	Number of components	4	5	6	4	5	6	4	5	6
ROI 2	Linear	54.03	53.71	54.35	52.58	52.1	52.42	52.74	53.39	55.32
	RBF	60.97	63.55	**65.32**	62.74	62.58	64.19	63.55	63.39	64.52
	JS	58.87	61.29	62.74	61.61	62.42	62.42	60.81	61.61	62.26
	JT	59.52	61.94	64.19	62.1	63.55	63.87	63.06	62.58	63.06
	WJT $\widetilde{k}_q$	59.84	61.77	63.71	**64.68**	65	64.52	64.52	**64.68**	64.52
	WJT k_q	59.52	61.29	62.74	65	64.52	64.35	64.52	64.35	64.35
ROI 4	Linear	60.81	60.48	59.52	57.9	58.87	57.9	58.39	59.03	58.23
	RBF	61.13	60.81	61.13	58.39	59.52	58.55	58.87	59.35	59.19
	JS	58.87	59.68	61.77	58.71	58.71	58.06	59.35	59.03	58.06
	JT	61.13	60.65	**62.26**	59.03	59.68	**59.84**	**60.65**	60.48	59.19
	WJT $\widetilde{k}_q$	61.65	61.94	61.94	58.87	59.19	58.71	59.68	59.19	59.03
	WJT k_q	59.52	60.16	61.94	58.55	59.03	58.23	59.03	59.03	59.84
ROI 6	Linear	57.1	58.23	59.19	56.45	57.1	58.23	58.23	58.55	58.87
	RBF	62.74	63.71	63.71	62.58	62.42	61.45	62.58	63.23	63.23
	JS	63.71	64.03	63.55	66.42	63.87	64.03	63.71	64.52	64.68
	JT	64.19	64.68	**65.48**	**66.61**	65.32	65.32	65.32	65.16	**65.81**
	WJT $\widetilde{k}_q$	64.84	65	65.16	62.42	63.22	64.03	62.26	63.06	63.71
	WJT k_q	64.19	64.68	**65.48**	63.06	62.74	64.03	62.26	64.03	64.03
ROI 8	Linear	60.16	60.32	60.32	59.68	59.52	59.35	60.16	61.45	60.16
	RBF	**67.26**	66.13	65	63.88	64.19	63.55	64.52	64.52	64.03
	JS	65.81	65.16	64.52	63.06	62.9	61.61	62.74	63.06	61.29
	JT	66.29	65.65	65	63.39	63.87	62.58	63.39	63.87	62.1
	WJT $\widetilde{k}_q$	66.13	65.65	64.84	64.84	65	64.68	65.16	65	64.35
	WJT k_q	66.29	65.32	65	65	**65.16**	64.52	**65.32**	65.16	64.35
ROI 12	Linear	59.03	59.35	59.35	57.9	58.39	58.55	58.71	58.55	57.9
	RBF	65.97	65.65	65.32	62.26	62.1	61.45	65.16	63.39	63.23
	JS	65.97	65.48	64.84	62.74	61.94	64.68	61.94	61.94	65
	JT	65.97	65.48	**66.45**	62.74	62.42	64.68	62.9	62.74	65.32
	WJT $\widetilde{k}_q$	66.13	65.48	**66.45**	65.32	63.06	65.48	65.48	63.23	64.68
	WJT k_q	65.97	65.48	**66.45**	**65.97**	64.84	65.32	**65.97**	65	65.16
ROI 14	Linear	55.32	55	55.48	55	54.84	54.84	55.65	55.65	56.13
	RBF	61.94	**62.74**	61.13	62.1	63.55	63.06	63.23	63.55	63.06
	JS	62.42	61.45	60.16	66.61	65.98	65.32	66.61	66.13	64.35
	JT	62.58	62.1	61.45	**67.9**	66.61	66.29	**68.06**	67.1	65.48
	WJT $\widetilde{k}_q$	**62.74**	61.94	61.45	65.48	64.84	63.87	65.48	65	63.87
	WJT k_q	62.58	62.1	61.45	65	64.19	63.71	64.68	64.68	63.23

Table 2
Mean accuracy for the best values of q and C for the SVM classifier learnt on ROI 10 using one Rician mixture per class with $K=4,5,6$ components and embeddings $\widetilde{\mathbf{e}}$, $\overline{\mathbf{e}}$ and $\widehat{\mathbf{e}}$.

	Embedding	$\widetilde{\mathbf{e}}$			$\overline{\mathbf{e}}$			$\widehat{\mathbf{e}}$		
	Number of components	4	5	6	4	5	6	4	5	6
ROI 10	Linear	55.97	56.29	56.45	56.29	56.94	54.84	55.81	56.61	56.29
	RBF	64.84	67.58	67.1	64.84	67.9	68.39	66.13	68.55	69.03
	JS	65.97	69.03	69.84	66.45	70	70.81	66.61	69.84	70.81
	JT	68.71	**71.77**	69.84	67.42	70.65	**71.13**	67.74	70	**71.61**
	WJT $\widetilde{k}_q$	68.55	71.29	70	65.81	69.19	**71.13**	65.81	70	70.48
	WJT k_q	68.71	**71.77**	69.84	65.65	68.71	70.48	65.65	70.16	70.65

of one mixture per class (schizophrenic and non-schizophrenic). We tested the generative embeddings $\widetilde{\mathbf{e}}$, $\overline{\mathbf{e}}$ and $\widehat{\mathbf{e}}$ proposed in Section 3, both in the single-mixture and R-mixtures versions.

The dataset contains 124 images (64 patients and 60 controls), each with the following 14 ROIs (7 pairs): Amygdala (1-Left, 2-Right), Dorso-lateral PreFrontal Cortex (3-Left, 4-Right), Entorhinal Cortex (5-Left, 6-Right), Heschl's Gyrus (7-Left, 8-Right), Hippocampus (9-Left, 10-Right), Superior Temporal Gyrus (11-Left, 12-Right), Thalamus (13-Left, 14-Right). To evaluate the classifiers, the dataset was split 50–50% into training and test subsets and 10 runs were performed.

SVM classifiers were trained for each individual ROI (without the boosting-based combination), and the conclusion was that ROI 10 leads to the best accuracy (see Tables 1 and 2—for each embedding, the best result is shown in boldface). The accuracy is robust to the number of components of the mixture. The best

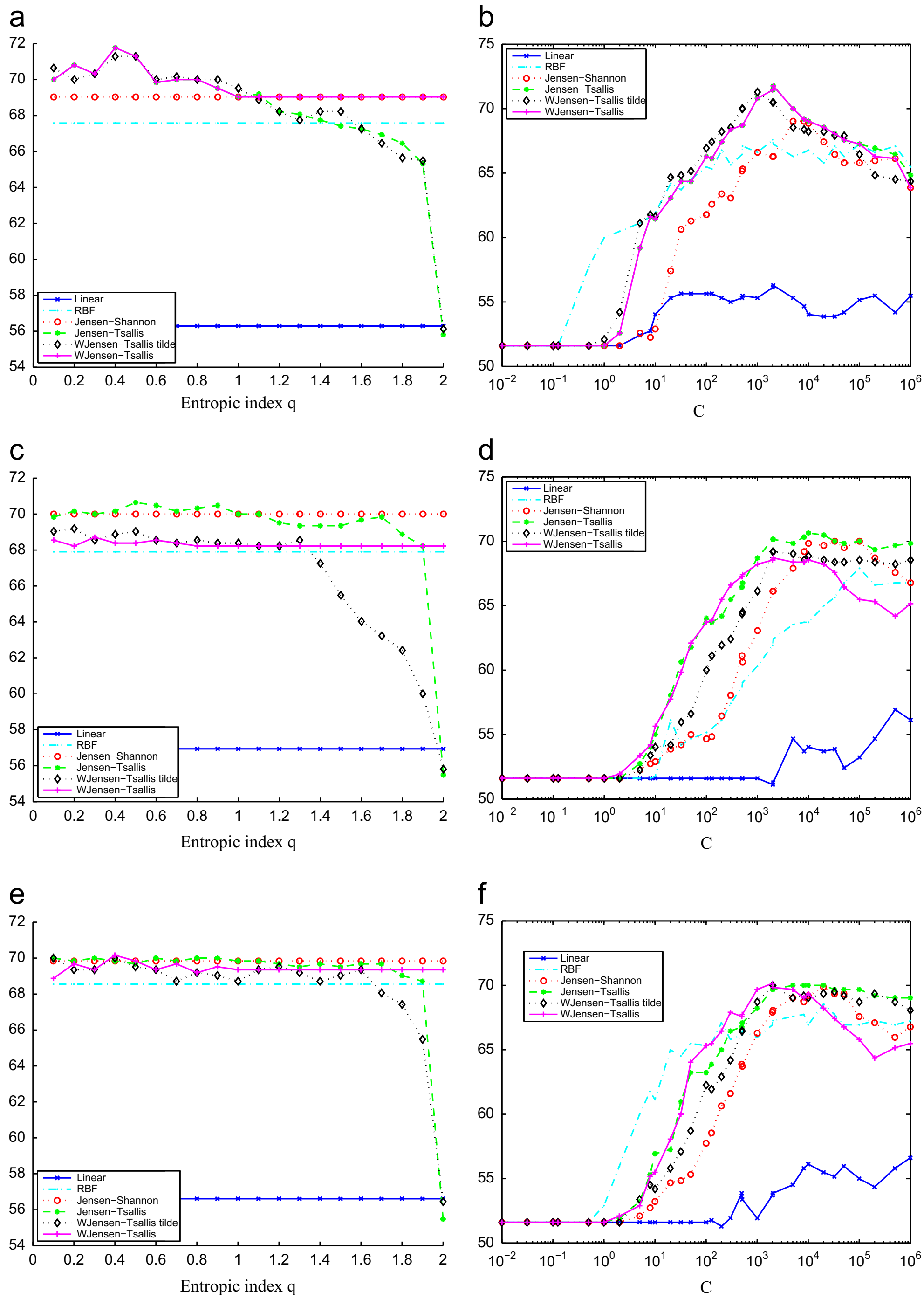

Fig. 2. Mean accuracy on 10 runs as a function of q (best C) and as a function of C (best q) for the SVM classifier learnt on ROI 10 using one Rician mixture per class with $K=5$ components and embeddings $\tilde{\mathbf{e}}$ ((a), (b)), $\bar{\mathbf{e}}$ ((c), (d)) and $\hat{\mathbf{e}}$ ((e), (f)).

performances over q and C are reported. For the GRBFK, the best performance over the width parameter and over C is reported. Mean accuracies are plotted in Fig. 2 as a function of q for the best value of C and as a function of C for the best value of q, for the generative embeddings $\tilde{\mathbf{e}}$, $\bar{\mathbf{e}}$ and $\hat{\mathbf{e}}$, with 2 (one per class) Rician mixtures each with 5 components. For $q>1$, the results shown for the weighted JT kernel (which is positive definite only for $q \in [0,1]$) correspond to $q=1$. These results show that the proposed generative embeddings lead to comparable performances. The information theoretic kernels outperform the LK and GRBFK. Namely, the best performances are obtained with the JT and weighted JT kernels, for all ROIs. The standard error of the mean is less than 0.006.

Results obtained by combining the SVM classifiers with the AdaBoost algorithm are shown in Table 3 for the generative embeddings $\tilde{\mathbf{e}}$, $\bar{\mathbf{e}}$ and $\hat{\mathbf{e}}$ (for each embedding, the best result is shown in boldface). These results show that the proposed approach outperforms state-of-the-art methods for ROIs intensity histograms for this dataset, see [16,40–42].

Results with a single estimated mixture for the entire dataset are similar. For both individual ROI and boosting experiments, the same considerations on embeddings and kernels performances as for the

Table 3
Mean accuracy for the best values of q and C for the set of SVM classifiers obtained by the boosting algorithm, using one Rician mixture per class with $K=4,5,6$ components and embeddings $\tilde{\mathbf{e}}$, $\bar{\mathbf{e}}$ and $\hat{\mathbf{e}}$. Results with state-of-the-art methods for ROIs intensity histograms using leave-one-out are also reported.

	Embedding	$\tilde{\mathbf{e}}$			$\bar{\mathbf{e}}$			$\hat{\mathbf{e}}$		
	Number of components	4	5	6	4	5	6	4	5	6
Boosting	JS	78.55	78.23	77.74	75	75.97	77.42	77.9	76.94	76.61
	JT	79.68	**80.16**	79.03	78.71	78.06	**79.84**	79.35	78.39	78.39
	WJT $\tilde{k}_q$	80	79.03	78.39	78.23	78.06	77.58	**81.77**	78.39	78.06
	WJT k_q	79.68	**80.16**	79.03	78.71	78.39	78.55	80.48	77.9	78.39

State-of-the-art methods

SVM best single ROI		*SVM multiple ROIs*	
Methodology	Accuracy	Methodology	Accuracy
[16]	73.4	Constellation probab.model + Fisher kernel [40]	80.65
Dissimilarity representations [42]	78.07	Combined dissimilarity representations [41]	79
		Dissimilarity representations [42]	76.32

Table 4
Mean accuracy for the best values of q and C for the SVM classifier learnt on ROI 10 using a single Rician mixture for the entire dataset with K=4, 5, 6 components and embeddings $\tilde{\mathbf{e}}$, $\bar{\mathbf{e}}$ and $\hat{\mathbf{e}}$.

	Embedding	$\tilde{\mathbf{e}}$			$\bar{\mathbf{e}}$			$\hat{\mathbf{e}}$		
	Number of components	4	5	6	4	5	6	4	5	6
ROI 10	Linear	55.81	56.13	56.64	56.77	55.32	54.84	56.29	56.77	55.81
	RBF	65	67.42	66.45	64.19	68.22	68.22	66.77	68.06	68.06
	JS	67.1	69.03	69.84	66.29	70.16	70.81	66.45	70.16	70.81
	JT	68.39	**70.48**	69.84	67.74	70.65	**71.45**	68.55	70.48	**71.29**
	WJT $\tilde{k}_q$	68.23	**70.48**	69.84	65.97	69.68	70.16	66.13	69.68	69.84
	WJT k_q	68.39	**70.48**	69.84	65.48	70	70.97	66.13	70	70.48

Table 5
Mean accuracy for the best values of q and C for the set of SVM classifiers obtained by the boosting algorithm, using a single Rician mixture for the entire dataset with $K=4,5,6$ components and embeddings $\tilde{\mathbf{e}}$, $\bar{\mathbf{e}}$ and $\hat{\mathbf{e}}$.

	Embedding	$\tilde{\mathbf{e}}$			$\bar{\mathbf{e}}$			$\hat{\mathbf{e}}$		
	Number of components	4	5	6	4	5	6	4	5	6
Boosting	JS	78.39	78.55	77.26	75.32	77.58	76.45	77.74	77.26	75.32
	JT	79.19	79.84	79.35	77.10	77.9	78.06	79.19	78.55	77.58
	WJT $\tilde{k}_q$	80.16	**80.48**	79.03	78.55	78.23	78.87	79.03	79.35	78.55
	WJT k_q	79.19	79.84	79.35	79.19	79.35	**79.52**	**80.16**	79.35	78.87

Table 6
Mean accuracy for the SVM classifier learnt on ROI 10 using one Rician mixture per class with $K=4,5,6$ components and embeddings $\tilde{\mathbf{e}}$, $\bar{\mathbf{e}}$ and $\hat{\mathbf{e}}$. The SVM C parameter is tuned by cross-validation over the training set. Results for the best value of q (*best q*) and for q tuned by cross-validation (q *cv*) are reported.

	Embedding	$\tilde{\mathbf{e}}$			$\bar{\mathbf{e}}$			$\hat{\mathbf{e}}$		
	Number of components	4	5	6	4	5	6	4	5	6
ROI 10	Linear	53.23	53.55	54.19	52.74	53.39	52.58	53.55	54.84	54.84
	RBF	58.55	61.61	63.39	60.32	61.77	65	61.29	64.84	62.9
	JS	65.16	67.26	65.48	63.06	67.58	**70.32**	64.52	66.13	**67.9**
	JT (q cv)	65.48	**66.94**	61.94	66.45	66.77	67.1	62.9	67.42	66.29
	WJT k_q (q cv)	65.81	66.13	62.58	58.39	65	66.94	62.42	65.16	67.58
	WJT $\tilde{k}_q$ (q cv)	64.35	66.13	62.9	64.52	65.16	66.45	64.03	65.32	**67.9**
	JT (best q)	65.81	68.23	65.48	66.29	68.55	**70.48**	67.26	67.42	**69.68**
	WJT $\tilde{k}_q$ (best q)	66.61	67.58	65.81	63.71	66.61	69.03	64.35	66.29	68.71
	WJT k_q (best q)	67.42	**68.39**	64.35	63.23	65.65	69.19	63.23	66.45	68.23

Table 7
Mean accuracy for the set of SVM classifiers obtained by the boosting algorithm, using one Rician mixture per class with $K=4,5,6$ components and embeddings $\tilde{\mathbf{e}}$, $\bar{\mathbf{e}}$ and $\hat{\mathbf{e}}$. The SVM C parameter is tuned by cross-validation over the training set. Results for the best value of q (*best q*) and for q tuned by cross-validation (q *cv*) are reported.

	Embedding	$\tilde{\mathbf{e}}$			$\bar{\mathbf{e}}$			$\hat{\mathbf{e}}$		
	Number of components	4	5	6	4	5	6	4	5	6
Boosting	JS	76.13	75	76.77	70.97	74.68	76.45	74.68	71.45	70.97
	JT (q cv)	75.81	77.26	76.77	75.81	74.68	77.9	76.61	74.68	76.94
	WJT $\tilde{k}_q$ (q cv)	74.19	**79.19**	78.87	76.61	74.19	75.65	75.65	71.77	72.58
	WJT k_q (q cv)	76.45	76.61	76.77	76.13	**78.23**	76.61	**78.06**	77.1	74.35
	JT (best q)	77.74	77.42	76.45	74.84	76.61	**80.48**	75.97	76.13	76.45
	WJT $\tilde{k}_q$ (best q)	76.45	**78.23**	77.1	75.32	76.61	77.1	76.94	75.65	76.13
	WJT k_q (best q)	77.74	77.26	75	79.84	77.42	75.97	76.94	**77.58**	76.13

case of 2 mixtures hold. For a single mixture, performances are lower, leading to a 71.45% accuracy as the best result in the case of ROI 10 and to a 80.48% accuracy as the best result in the case of boosting. Results for a single estimated mixture are reported in Tables 4 and 5 (for each embedding, the best result is shown in boldface).

6.1. Cross-validation results

Experiments were also performed with the parameters tuned by cross-validation over the training set. Performances of the kernels were computed as a function of the entropic index q, with the SVM C parameter tuned by cross-validation over the training set, leading to a 70.48% accuracy as the best result for ROI 10 and to a 80.48% accuracy as the best result for boosting. Cross-validation results are reported in Tables 6 and 7 (for each embedding, the best result is shown in boldface).

7. Conclusions

In this paper, we have proposed a new approach for building generative embeddings for kernel-based classification of magnetic resonance images (MRI) by exploiting the Rician distribution that characterizes MR images. Using generative embeddings, the images to be classified are mapped onto a Hilbert space, where kernel-based techniques can be used. Concerning the choice of kernel, we have adopted the recently proposed nonextensive information theoretic kernels. The proposed approach was tested on a challenging classification task: classifying subjects as suffering, or not, from schizophrenia on the basis of a set of regions of interest (ROIs) in each image. For this purpose, an SVM classifier for each ROI is learnt. Finally, we propose to combine the SVM classifiers via a boosting algorithm. The experimental results show that the proposed methodology outperforms the previous state-of-the-art methods on the same dataset. At a more general level, we may claim that our results contribute to the conclusion that the combination of generative embeddings with information theoretic kernels is a competitive approach for challenging classification problems.

Acknowledgments

This work was partially supported by the Fundação para a Ciência e a Tecnologia (Portugal), under Project PEst-OE/EEI/LA0008/2011.

Appendix A. Proof of Proposition 2.1

Proof. First of all, let us note that $f(y_j;\boldsymbol{\theta}_i)$ can be written in factorized form as

$$f_i(y_j;\boldsymbol{\theta}_i)=A(y_j;\boldsymbol{\theta}_i)\cdot B(y_j;\boldsymbol{\theta}_i), \tag{A.1}$$

where

$$A(y_j;\boldsymbol{\theta}_i)=\frac{y_j}{\sigma_i^2}e^{-(y_j^2+v_i^2)/2\sigma_i^2} \tag{A.2}$$

and

$$B(y_j;\boldsymbol{\theta}_i)=I_0\left(\frac{y_j v_i}{\sigma_i^2}\right). \tag{A.3}$$

It follows that the partial derivatives of the log-likelihood with respect to v_i and σ_i^2 result

$$\frac{\partial \log f(y_j;\boldsymbol{\theta}_i)}{\partial v_i}=\frac{1}{A\cdot B}\cdot\left[\frac{\partial A}{\partial v_i}\cdot B+A\cdot\frac{\partial B}{\partial v_i}\right]=\frac{1}{A}\cdot\frac{\partial A}{\partial v_i}+\frac{1}{B}\cdot\frac{\partial B}{\partial v_i} \tag{A.4}$$

$$\frac{\partial \log f(y_j;\boldsymbol{\theta}_i)}{\sigma_i^2}=\frac{1}{A}\cdot\frac{\partial A}{\partial\sigma_i^2}+\frac{1}{B}\cdot\frac{\partial B}{\partial\sigma_i^2}. \tag{A.5}$$

The partial derivative of $A(y_j;\boldsymbol{\theta}_i)$ with respect to v_i is

$$\frac{\partial A(y_j;\boldsymbol{\theta}_i)}{\partial v_i}=\frac{y_j}{\sigma_i^2}e^{-(y_j^2+v_i^2)/2\sigma_i^2}\cdot\left(-\frac{1}{2\sigma_i^2}\cdot 2v_i\right). \tag{A.6}$$

Moreover, recalling that the higher order modified Bessel functions $I_n(z)$, defined by the contour integral

$$I_n(z)=\frac{1}{2\pi i}\oint e^{(z/2)((t+1)/t)}t^{-n-1}\,dt, \tag{A.7}$$

where the contour encloses the origin and is traversed in a counterclockwise direction, can be expressed in terms of $I_0(z)$ through the following derivative identity [28]:

$$I_n(z)=T_n\left(\frac{d}{dz}\right)I_0(z) \tag{A.8}$$

where $T_n(z)$ is a Chebyshev polynomial of the first kind [28]

$$T_n(z)=\frac{1}{4\pi i}\oint\frac{(1-t^2)t^{-n-1}}{(1-2tz+t^2)}\,dt, \tag{A.9}$$

with the contour enclosing the origin and traversed in a counterclockwise direction, and in particular that $T_1(z)=z$, then the partial derivative of B results

$$\frac{\partial B(y_j;\boldsymbol{\theta}_i)}{\partial v_i}=\frac{\partial I_0\left(\frac{y_j v_i}{\sigma_i^2}\right)}{\partial v_i}=I_1\left(\frac{y_j v_i}{\sigma_i^2}\right)\cdot\frac{y_j}{\sigma_i^2}. \tag{A.10}$$

Substituting (A.6) and (A.10) in (A.4) we get

$$\frac{\partial \log f(y_j;\boldsymbol{\theta}_i)}{\partial v_i} = -\frac{v_i}{\sigma_i^2} + \frac{I_1\left(\frac{y_j v_i}{\sigma_i^2}\right)}{I_0\left(\frac{y_j v_i}{\sigma_i^2}\right)} \cdot \frac{y_j}{\sigma_i^2} \tag{A.11}$$

which, substituted in (9) yields (10).

The same considerations hold for the partial derivatives with respect to σ_i^2, yielding to the following expressions for the partial derivative of A and B (with respect to σ_i^2)

$$\frac{\partial A(y_j;\boldsymbol{\theta}_i)}{\partial \sigma_i^2} = -\frac{y_j}{\sigma_i^4} e^{-(y_j^2+v_i^2)/2\sigma_i^2} + \frac{y_j}{\sigma_i^2} e^{-(y_j^2+v_i^2)/2\sigma_i^2} \frac{y_j^2+v_i^2}{2\sigma_i^4} \tag{A.12}$$

$$\frac{\partial B(y_j;\boldsymbol{\theta}_i)}{\partial \sigma_i^2} = I_1\left(\frac{y_j v_i}{\sigma_i^2}\right) \cdot \frac{y_j v_i}{\sigma_i^4} \tag{A.13}$$

Substituting (A.12) and (A.13) in (A.5), the partial derivative of $\log f(y_j;\boldsymbol{\theta}_i)$ with respect to σ_i^2 results

$$\frac{\partial \log f(y_j;\boldsymbol{\theta}_i)}{\partial \sigma_i^2} = -\frac{1}{\sigma_i^2}\left(1 - \frac{y_j^2+v_i^2}{2\sigma_i^2}\right) - \frac{I_1\left(\frac{y_j v_i}{\sigma_i^2}\right)}{I_0\left(\frac{y_j v_i}{\sigma_i^2}\right)} \cdot \frac{y_j v_i}{\sigma_i^4}$$

which, plugged in (9) yields (11). □

References

[1] A. Ng, M. Jordan, On discriminative vs generative classifiers: a comparison of logistic regression and naïve Bayes, in: Advances in Neural Information Processing Systems, 2002.

[2] Y.D. Rubinstein, T. Hastie, Discriminative vs informative learning, in: Proceedings of the 3rd International Conference on Knowledge Discovery and Data Mining, 1997, pp. 49–53.

[3] T. Jaakkola, D. Haussler, Exploiting generative models in discriminative classifiers, in: Advances in Neural Information Processing Systems, vol. 11, 1999, pp. 487–493.

[4] J. Lasserre, C. Bishop, T. Minka, Principled hybrids of generative and discriminative models, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2006, pp. 87–94.

[5] D. Haussler, Convolution Kernels on Discrete Structures, Technical Report UCSC-CRL-99-10, Department of Computer Science, University of California, Santa Cruz, 1999.

[6] C. Watkins, Dynamic alignment kernels, in: Advances in Large Margin Classifiers, 1999, pp. 39–50.

[7] K. Tsuda, T. Kin, K. Asai, Marginalized kernels for biological sequences, Bioinformatics 18 (2002) 268–275.

[8] P.J. Moreno, P.P. Ho, N. Vasconcelos, A Kullback–Leibler divergence based kernel for SVM classification in multimedia applications, in: Advances in Neural Information Processing Systems, 2003.

[9] R. Kondor, T. Jebara, A kernel between sets of vectors, in: Proceedings of the International Conference on Machine Learning, 2003, pp. 361–368.

[10] T. Jebara, R. Kondor, A. Howard, Probability product kernels, J. Mach. Learn. Res. 5 (2004) 819–844.

[11] M. Hein, O. Bousquet, Hilbertian metrics and positive definite kernels on probability measures, in: Proceedings of the International Conference on Artificial Intelligence and Statistics, 2005, pp. 136–143.

[12] M. Cuturi, J. Vert, Semigroup kernels on finite sets, in: Advances in Neural Information Processing Systems, vol. 17, 2005, pp. 329–336.

[13] M. Cuturi, K. Fukumizu, J. Vert, Semigroup kernels on measures, J. Mach. Learn. Res. 6 (2005) 1169–1198.

[14] A.F.T. Martins, N.A. Smith, P.M. Aguiar, M.A.T. Figueiredo, Nonextensive information theoretic kernels on measures, J. Mach. Learn. Res. 10 (2009) 935–975.

[15] C. Tsallis, Possible generalization of Boltzmann–Gibbs statistics, J. Stat. Phys. 52 (1988) 479–487.

[16] D. Cheng, M. Bicego, U. Castellani, S. Cerutti, M. Bellani, G. Rambaldelli, M. Atzori, P. Brambilla, V. Murino, Schizophrenia classification using regions of interest in brain MRI, in: Intelligent Data Analysis in Biomedicine and Pharmacology Workshop, 2009.

[17] H. Gudbjartsson, S. Patz, The Rician distribution of noisy MRI data, Magn. Reson. Med. 34 (1995) 910–914.

[18] R.M. Henkelman, Measurement of signal intensities in the presence of noise in MR images, Med. Phys. 12 (1985) 232–233.

[19] S.O. Rice, Mathematical analysis of random noise, Bell Syst. Tech. J. 24 (1945) 46–156.

[20] B.P. Lathi, Modern Digital and Analog Communication Systems, Hault-Saunders International Edition, 1983.

[21] R. Maitra, D. Faden, Noise estimation in magnitude MR datasets, IEEE Trans. Med. Imaging 28 (2009) 1615–1622.

[22] R. Maitra, J. Riddles, Synthetic magnetic resonance imaging revisited, IEEE Trans. Med. Imaging 29 (2010) 895–902.

[23] S. Roy, A. Carass, P. Bazin, J. Prince, A Rician mixture model classification algorithm for magnetic resonance images, in: Proceedings of the IEEE International Symposium on Biomedical Imaging, 2009, pp. 406–409.

[24] Y. Freund, R. Schapire, A decision-theoretic generalization of online learning and an application to boosting, J. Comput. Syst. Sci. 55 (1997) 119–139.

[25] A.C. Carli, M.A.T. Figueiredo, M. Bicego, V. Murino, Generative embeddings based on Rician mixtures: application to kernel-based discriminative classification of magnetic resonance images, in: Proceedings of the First International Conference on Pattern Recognition Applications and Methods, 2012, pp. 113–122.

[26] A. Chung, J. Noble, Statistical 3D vessel segmentation using a Rician distribution, in: Proceedings of the International Conference on Medical Image Computing and Computer Assisted Intervention, 1999, pp. 82–89.

[27] R. Maitra, On the expectation–maximization algorithm for Rice–Rayleigh mixtures with application to noise parameter estimation in magnitude MR datasets, Sankhyá: The Indian J. Stat. 75 (2013).

[28] M. Abramowitz, I. Stegun, Handbook of Mathematical Functions, Dover, New York, 1972.

[29] A. Dempster, N. Laird, D. Rubin, Maximum likelihood from incomplete data via the EM algorithm, J. R. Stat. Soc. (B) 39 (1977) 1–38.

[30] G. McLachlan, D. Peel, Finite Mixture Models, John Wiley & Sons, New York, 2000.

[31] G. McLachlan, T. Krishnan, The EM Algorithm and Extension, John Wiley & Sons, New York, 2006.

[32] C. Fraley, A.E. Raftery, Model-based clustering, discriminant analysis, and density estimation, J. Am. Stat. Assoc. 97 (2002) 611–631.

[33] V. Melnykov, R. Maitra, Finite mixture models and model-based clustering, Stat. Surv. 4 (2010) 80–116.

[34] H. Suyari, Generalization of Shannon–Khinchin axioms to nonextensive systems and the uniqueness theorem for the nonextensive entropy, IEEE Trans. Inf. Theory 50 (2004) 1783–1787.

[35] J. Burbea, C. Rao, On the convexity of some divergence measures based on entropy functions, IEEE Trans. Inf. Theory 28 (1982) 489–495.

[36] J. Lin, Divergence measures based on Shannon entropy, IEEE Trans. Inf. Theory 37 (1991) 145–151.

[37] N. Cristianini, J. Shawe-Taylor, An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, 2000.

[38] B. Schölkopf, A.J. Smola, Learning with Kernels, MIT Press, 2002.

[39] M.A.T. Figueiredo, A.K. Jain, Unsupervised learning of finite mixture models, IEEE Trans. Pattern Anal. Mach. Intell. 24 (2002) 381–396.

[40] D. Cheng, M. Bicego, U. Castellani, M. Cristani, S. Cerruti, M. Bellani, G. Rambaldelli, M. Aztori, P. Brambilla, V. Murino, A hybrid generative/discriminative method for classification of regions of interest in schizophrenia brain MRI, in: MICCAI 2009 Workshop on Probabilistic Models for Medical Image Analysis, 2009.

[41] A. Ulas, R. Duin, U. Castellani, M. Loog, M. Bicego, V. Murino, M. Bellani, S. Cerruti, M. Tansella, P.Brambilla, Dissimilarity-based detection of schizophrenia, in: ICPR Workshop on Brain Decoding: Pattern Recognition Challenges in fMRI Neuroimaging, 2010, pp. 32–35.

[42] A. Ulas, R. Duin, U. Castellani, M. Loog, P. Mirtuono, M. Bicego, V. Murino, M. Bellani, S. Cerruti, M. Tansella, P. Brambilla, Dissimilarity-based detection of schizophrenia, Int. J. Imaging Syst. Technol. 21 (2011) 179–192.

Anna Caterina Carli received the M.Sc. degree (with honours) in Telecommunication Engineering from Politecnico di Milano, Milan, Italy, in 2003, and the Ph.D. degree in Computer Science from University of Verona, Italy, in 2012. From January to May 2010, she was a visiting Ph.D. student at Instituto de Telecomunicações and the Department of Electrical and Computer Engineering, Instituto Superior Técnico, Technical University of Lisbon, Portugal. She currently carries out research activities at the Dipartimento per le Comunicazioni, Ministero dello Sviluppo Economico, Rome, Italy. Her current research interests include pattern recognition, statistical learning (in particular kernel methods), optimization, and information theory.

Mário A.T. Figueiredo received his Ph.D. degree in Electrical and Computer Engineering, from Instituto Superior Técnico (IST), the Engineering School of the Technical University of Lisbon (TULisbon), Portugal, in 1994. Since 1994, he has been with the faculty of the Department of Electrical and Computer Engineering, IST, where he is now a full professor. He is also an area coordinator at Instituto de Telecomunicações, a private not-for-profit research institution. His research interests include image processing and analysis, pattern recognition, statistical learning, and optimization. M. Figueiredo is a fellow of the IEEE (Institute of Electrical and Electronics Engineers) and of the IAPR (International

Association for Pattern Recognition) and was a member of the Image, Video, and Multidimensional Signal Processing Technical Committee of the IEEE. He received the 1995 Portuguese IBM Scientific Prize, the 2008 UTL/Santander-Totta Scientific Prize, and co-authored a paper that received the 2011 IEEE Signal Processing Best Paper Award. He is/was an associate editor of the following journals: IEEE Transactions on Image Processing, IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE Transactions on Mobile Computing, Pattern Recognition Letters, Signal Processing, Statistics and Computing. He was a co-chair of the 2001 and 2003 Workshops on Energy Minimization Methods in Computer Vision and Pattern Recognition, and program/technical committee member of many international conferences.

Manuele Bicego received his Laurea degree and Ph.D. degree in Computer Science from University of Verona in 1999 and 2003, respectively. From 2004 to 2008 he was at the University of Sassari, in the Computer Vision Lab. Currently he is an assistant professor (ricercatore) at the University of Verona, and member of the VIPS (Vision Image Processing & Sound) lab at the Computer Science Department. From June 2009 to February 2011 he was also a member of the PLUS (Pattern analysis, Learning and image Understanding Systems) lab at the Istituto Italiano di Tecnologia (IIT, Genova, Italy). His research interests include statistical pattern recognition, mainly probabilistic models (GMM, HMM) and kernel machines (e.g., SVM), with application to video analysis, biometrics and, recently, bioinformatics. Manuele Bicego is an author of several papers in the above subjects, published in international journals and conferences. Since 2004 he is an associate editor of the international journal Electronic Letters on Computer Vision and Image Analysis (ELCVIA); he was a guest editor of the special issue of Pattern Recognition on “Similarity Based Pattern Recognition” (vol. 39(10), 2006). From 2008 he is an associate editor of the International Journal of Imaging. He has served as a member of the scientific committee of different international conferences, and he is a reviewer for several international conferences and journals. Manuele Bicego is a member of the IEEE Systems, Man, and Cybernetics society and of the IAPR Society Italian Chapter (GIRPR).

Vittorio Murino is a full professor at the University of Verona, Italy, and director of the PAVIS (Pattern Analysis and Computer Vision) Department at the Istituto Italiano di Tecnologia. He took the Laurea degree in Electronic Engineering in 1989 and the Ph.D. in Electronic Engineering and Computer Science in 1993 at the University of Genova, Italy. He held a post-doctoral position from 1993 to 1995, working in the Signal Processing and Understanding Group of the Department of Biophysical and Electronic Engineering of the University of Genova as a supervisor of research activities on image processing for object recognition and pattern classification in underwater environments. From 1995 to 1998, he was an assistant professor at the Department of Mathematics and Computer Science of the University of Udine, Italy, and since 1998 he works at the University of Verona. Here, he was among the main contributors for the foundation of the Department of Computer Science, for which also served as a chairman from 2001 to 2007. Prof. Murino is scientific responsible of several national and European projects, and evaluator of EU project proposals related to several frameworks and programs. Currently, he is working at the Istituto Italiano di Tecnologia in Genova, Italy, to set up and lead the PAVIS facility involved in computer vision, machine learning, and image analysis issues. His main research interests include: computer vision and pattern recognition/machine learning, in particular, probabilistic techniques for image and video processing, with applications on video surveillance, biomedical image analysis and bioinformatics. Prof. Murino is a co-author of about 250 papers published in refereed journals and international conferences, reviewer for several international journals, and member of the technical committees of important conferences (CVPR, ICCV, ECCV, ICPR, ICIP and others), and guest co-editor of special issues in relevant scientific journals. He is also a member of the editorial board of Pattern Recognition, Pattern Analysis and Applications, and Machine Vision & Applications journals, as well as of the IEEE Transactions on Systems Man, and Cybernetics Part B: Cybernetics. Finally, Prof. Murino is a senior member of the IEEE and Fellow of the IAPR.