Generative Adversarial Networks and Markov Random Fields for oversampling very small training sets

Adisson Salazar, Luis Vergara, Gonzalo Safont

Universitat Politècnica de València, Camino de Vera s/n, 46022, València, SPAIN

e-mail: asalazar@dcom.upv.es

Abstract

In this work, we propose a new method for oversampling the training set of a classifier, in a scenario of extreme scarcity of training data. It is based on two concepts: Generative Adversarial Networks (GAN) and vector Markov Random Field (vMRF). Thus, the generative block of GAN uses the vMRF model to synthesize surrogates by the Graph Fourier Transform. Then, the discriminative block implements a linear discriminant on features measuring clique similarities between the synthesized and the original instances. Both blocks iterate until the linear discriminant cannot discriminate the synthetic from the original instances. We have assessed the new method, called Generative Adversarial Network Synthesis for Oversampling (GANSO), with both simulated and real data in experiments where the classifier is to be trained with just 3 or 5 instances. The applications consisted of classification of stages of neuropsychological tests using electroencephalographic (EEG) and functional magnetic resonance imaging (fMRI) data and classification of sleep stages using electrocardiographic (ECG) data. We have verified that GANSO can effectively improve the classifier performance, while the benchmark method SMOTE is not appropriate to deal with such a small size of the training set.

Keywords: classifier training, oversampling, generative adversarial networks, Markov random fields

1. Introduction

Scarcity of data is a classical issue in the design and testing of automatic classifiers. The most typical scenario is that of imbalance (Guo et al., 2017; Krawczyk, 2016; López et al., 2013; He & García, 2009) are some representative reviews of the many ones existing to this regard. Detection of credit card frauds (Bhattacharyya, 2011; Salazar, 2018) or non-destructive testing of materials (Liao, 2008) can be significantly different application domains for the imbalance case. But a more general setting also includes those applications where there is a lack of data for all the assumed classes. For example, automatic classification of a variety of pathologies from the analysis of biomedical signals or images (Jie et al., 2018; Beleites et al., 2013) requires a large number of captures from a large number of patients, or from the same patient.

Oversampling of the defective classes is one obvious option to alleviate the scarcity of training/testing data. In the data processing community, different variants around the main idea of interpolating the available original data have been proposed. Interpolation methods are simple to implement and have general applicability: they do not require assuming any statistical model. The most consolidated technique is the Synthetic Minority Oversampling Technique (SMOTE), first proposed in 2002 (Chawla el al., 2002). A given number of synthetic instances are obtained from every original instance by random interpolation with some selected neighbors. This is made in an effort to preserve the local properties of the implicit multivariate probability density function (MPDF) in the oversampled space. Since 2002, different variants have appeared around this main idea (Fernández et al., 2018), like Borderline-SMOTE (Han et al., 2005), Adaptive Synthetic (ADASYN) (He et al., 2008) and Self-level-SMOTE (Bunkhumpornpat et al., 2009), among others. Basically, these extensions of SMOTE relate on an unequal consideration of every original instance: more attention is payed to the instances that are closer to the instances of the other classes, i.e., (more formally) that are closer to the overlapping area between the implicit MPDFs of the different classes. Unfortunately, the statistical characterization given by a very small number of original instances is poor, so recovering the implicit MPDF by interpolation is not possible. We may find a similarity with recovering a signal by interpolation of the available samples (sampling theorem). We will see in the experimental section that SMOTE is not appropriate for the very small number of samples considered in this research.

Moreover, in the statistical signal processing area, one can find a diversity of synthesis methods which try to (approximately) sample from some MPDF (see for example Angeletti, Bertin & Abry, 2013). Synthesizing by sampling from some MPDF allows replication of statistical properties of the original signals. However direct parametric or non-parametric estimation of the MPDF requires a large number of training instances. Even assuming perfect knowledge of the MPDF, sampling from it cannot be a simple task except for very specific types of densities.

In this work, we propose a new approach based on two concepts: Generative Adversarial Networks (GAN) (Goodfellow, 2016; Lin et al., 2018; Su et al., 2019; Li et al., 2020), an emerging paradigm in machine learning, and vector Markov Random Field (vMRF), an extension of the classical MRF (Chellappa & Jain, 1991). As we will see, the method may be considered an effort to incorporate the merits of the two mentioned approaches: no explicit estimation of the MPDF is required, but structural information of the original data can be incorporated into the synthetic data. Other recent approach also incorporates some type of structural information to alleviate the scarcity of training data: Few-Shot Learning (FSL) (Lake, 2011; Han, 2018). FSL is inspired by the form that humans learn new classes from a few (even only one) representative instance. This learning takes advantage from some prior structural high-level information (in the case of humans derived from the historic brain learning process). Thus, for example, a new animal class can be learned from just a few shots because there is a prior knowledge about the essential structure of an animal (head, body, legs …). Stated in a more formal manner, there is assumed a structural model whose parameters can be learned from a few outputs of the model. FSL has been applied to complex instances like images or written characters, exploiting complex high-level models. In contrast, our approach exploits much simpler structural information in the form of a vMRF which connects segments of the instances that are assumed to be correlated. Moreover, this prior information is not directly used to separate the different classes but to improve the synthetic instances obtained from the original ones. Thus, different classes could share the same vMRF used to oversample their respective instance space. The proposed approach was tested in the following real applications: classification of stages of a neuropsychological test (Barcelona test) using electroencephalographic (EEG) data from epileptic patients; classification of stages of a neuropsychological test (1-back working memory task) using functional Magnetic Resonance Images (fMRI) from individuals with schizophrenia; and classification of sleep stages using electrocardiographic (ECG) data from apnea patients.

2. Preliminaries

2.1 Generative adversarial network

Let $\mathbf{x} \in \square^{M}$ a vector that represent random instances defined by some unknown probability density $p(\mathbf{x})$. We propose an indirect sampling method, without requiring explicit knowledge of $p(\mathbf{x})$. It is based on a GAN structure. The GAN is composed by two blocks. The first one is generative and tries to generate synthetic instances $\mathbf{s}$ so that $p(\mathbf{s}) \square p(\mathbf{x})$. The second is a discriminative detector, which tries to discriminate the original instances from the synthetic ones provided by the generative block. Convergence is reached when the discriminative block cannot effectively distinguish original from synthetic instances (i.e., a systematic posterior probability close to 0.5 is assigned to both classes). This condition will be considered as indicative of the generative block sampling from the correct distribution.

2.2 Vector Markov Random Field

In principle any discriminator and any generator implementation are candidates for the GAN, but this implies too arbitrariness. This is because we face an ill-conditioned problem: many possible $p(\mathbf{x})$ can be compatible with a reduced number of available original instances. So we need some kind of regularization (in a wide sense) to constraint the properties that are to be prioritized in the synthetic instances. Regularization may emanate from some structural assumptions about $\mathbf{x}$, derived from prior knowledge (eg., expert informed data) or by standard analysis methods in specific application domains. Structured approach leads naturally to the consideration of MRF as we show in the following.

Let us consider that $\mathbf{x}$ is segmented in L non-overlapping segments $\mathbf{x}=\left[\mathbf{x}_1^T...\mathbf{x}_L^T\right]^T$. We assume that **x** is a vMRF over an undirected graph $G(V, E, \mathbf{A})$, where V is a set of L vertices, E is the set of edges joining vertices and **A** is the (LxL) symmetric adjacency matrix. A vMRF is a generalization of a MRF, assuming that vectors rather than scalars are assigned to the graph vertices. Thus, every segment $\mathbf{x}_l$ is assigned to vertex l, so that dependent segments are connected by an edge equal to 1, i.e., the corresponding elements in **A** are set to 1, while independent segments are left disconnected, i.e., the corresponding elements in **A** are set to 0. The fundamental theorem of MRF can be straightforwardly extended to vMRF, so that we can factorize $p(\mathbf{x})$ in the form

$$p(\mathbf{x})=\prod_{c=1}^{C} p\left(\mathbf{x}^c\right) \qquad . \qquad (1)$$

Where c runs over the maximal cliques: subsets of fully connected vertices that cannot be extended by adding more adjacent vertices. Thus, naming $\mathbf{x}_i^c$ to the i-th segment of the c-th maximal clique, we can write $\mathbf{x}^c=\left[\left(\mathbf{x}_1^c\right)^T...\left(\mathbf{x}_{L_c}^c\right)^T\right]^T$ where $\left\{\mathbf{x}_1^c...\mathbf{x}_{L_c}^c\right\}\subseteq\left\{\mathbf{x}_1...\mathbf{x}_L\right\}$. Notice that one segment can be a member of more than one maximal clique, hence $\sum_{c=1}^{C} L_c \geq L$. Also notice that a vMRF can be interpreted as a simplification of a Conditional Random Field (CRF) (Lafferty, McCallum & Pereira, 2001; Perez-Cruz, Pontil & Ghahramani, 2007). In a CRF every segment $\mathbf{x}_l$ (input) is tagged with a discrete scalar y_l (output) and a joint factorization is possible $p(\mathbf{y},\mathbf{x})=\prod_{c=1}^{C} p\left(\mathbf{y}^c,\mathbf{x}^c\right)$. CRF are assumed to implement discriminative classifiers to deduce the most probable tags from an observed **x**. Thus, the goal is to learn $P(\mathbf{y}/\mathbf{x})=p(\mathbf{y},\mathbf{x})/p(\mathbf{x})=\left(\prod_{c=1}^{C} p\left(\mathbf{y}^c,\mathbf{x}^c\right)\right)\Big/p(\mathbf{x})$. But in our case the output **y** is of no concern or could be considered to be included in the own vector **x**. The essential aspect is that factorization (1) is assumed to model the relevant structural information to be preserved in the synthetic instances.

The next section is dedicated to present the new oversampling algorithm. Then section 4 is devoted to experiments with simulated and real data for assessing the practical interest of the proposed synthesis method.

3. The GAN Synthesis for Oversampling method

Figure 1 depicts the essential aspects of the method. White arrows indicate the input information required by every block, while solid arrows indicate the outputs they provide.

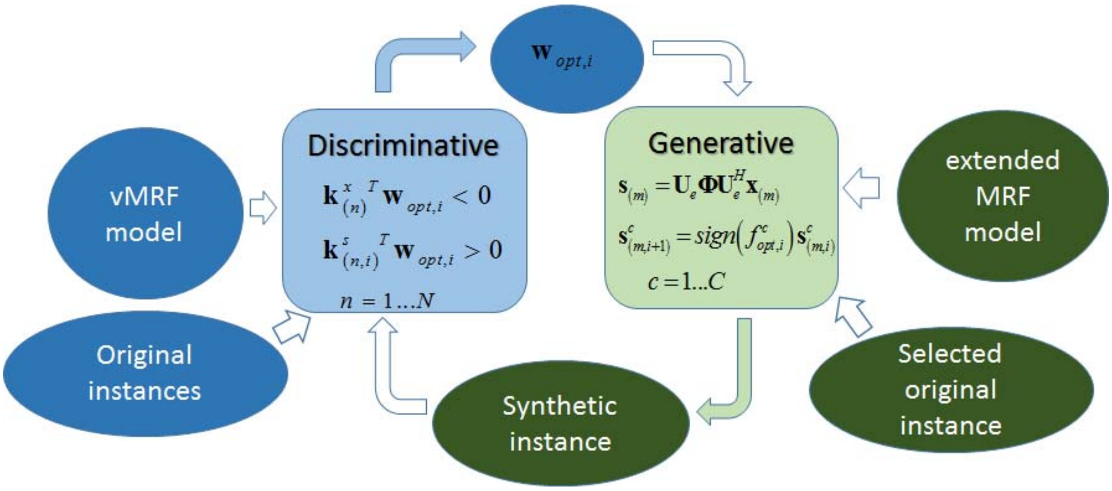
$\mathbf{w}_{opt,i}$
vMRF model
Discriminative
$\mathbf{k}^{x}_{(n)}{}^{T}\mathbf{w}_{opt,i} < 0$
$\mathbf{k}^{s}_{(n,i)}{}^{T}\mathbf{w}_{opt,i} > 0$
$n = 1\ldots N$
Generative
$\mathbf{s}_{(m)} = \mathbf{U}_e \mathbf{\Phi} \mathbf{U}_e^H \mathbf{x}_{(m)}$
$\mathbf{s}^c_{(m,i+1)} = sign\left(f^c_{opt,i}\right)\mathbf{s}^c_{(m,i)}$
$c = 1\ldots C$
extended MRF model
Original instances
Synthetic instance
Selected original instance

Figure 1. GAN scheme of the proposed synthesis method

Thus, the generative block requires knowledge of the extended MRF and the selected original instance from which a synthetic one is to be generated and given to the discriminative block for testing its validity. It also requires knowledge of the current optimum discrimination coefficients to correct the synthetic instances so that the discriminator could be "cheated". On the other hand, the discriminative block requires knowledge of the vMRF model, the original instances, and the synthetic instance under test. With this information the optimum discriminative coefficients are updated. If the synthetic instance under test can be discriminated from the set of original instances, the optimum coefficients are provided to the generative block to improve the synthesis.

In the following sections we make a detailed description of every block.

3.1 The generative block

To generate a synthetic instance we follow a surrogating approach. Surrogates of a given time signal are computed by Fourier transforming it to a spectral domain where the magnitude is preserved while the phase is randomized. Then the inverse Fourier transform is computed. Preserving the spectral magnitude guarantees that the covariance properties of the time signal are kept in the surrogates. Eventually, some corrections can be made in the original domain to preserve some additional properties like the empirical amplitude. Surrogating has been proposed as a general method of signal synthesis (Borgnat, Abry & Flandrin, 2012), although the classical application is that of hypothesis testing to decide if the signal fits or not some prescribed models (Miralles et al., 2008; Mandic et al., 2008). Recently, this approach has been extended to arbitrary domains by considering the Graph Fourier Transform (GFT) (Pirondini et al., 2016, Vergara et al., 2017, Belda et al., 2019). This is of special interest in our case, as preserving the magnitude of the GFT guarantees that the graph connectivity properties of the original signal (as defined by the vMRF model) are preserved in the graph signal surrogate. Hence, let us describe in the following the proposed method to implement the generative block.

First, let us define the concept of "extended undirected graph" $G(V_e,E_e,\mathbf{A}_e)$. This corresponds to an extension of the graph $G(V,E,\mathbf{A})$ where every sample of $\mathbf{x}$ is assigned to a vertex of the set V_e. This vertices are connected by the set of edges E_e in the form defined by the extended symmetric adjacency matrix $\mathbf{A}_e$. This matrix is defined so that all vertices corresponding to samples from the same segment or from connected segments in $G(V,E,\mathbf{A})$ are connected. Otherwise, the samples are left disconnected. So $\mathbf{A}_e$ is a block matrix formed by a total number of L^2 blocks

$$\mathbf{A}_e = \begin{pmatrix} \mathbf{A}_{e11} & \cdots & \mathbf{A}_{e1L} \\ \vdots & \ddots & \vdots \\ \mathbf{A}_{eL1} & \cdots & \mathbf{A}_{eLL} \end{pmatrix} \qquad \mathbf{A}_{eij} = \begin{pmatrix} a_{ij} & \cdots & a_{ij} \\ \vdots & \ddots & \vdots \\ a_{ij} & \cdots & a_{ij} \end{pmatrix} \qquad , \qquad (2)$$

Where a_{ij}=1 if segments i and j are connected in $G(V,E,\mathbf{A})$ and a_{ij}=0 if they are disconnected.

Then let us generate a surrogate of some original instance $\mathbf{x}_{(n)}$. First, we compute the GFT

$$GFT\left(\mathbf{x}_{(n)}\right) = \mathbf{U}_e^H \mathbf{x}_{(n)} = \mathbf{r}_{(n)} \qquad , \qquad (3)$$

the columns of $\mathbf{U}_e$ are the eigenvectors of the extended graph Laplacian matrix, $\mathbf{L}_e = \mathbf{D}_e - \mathbf{A}_e$, being $\mathbf{D}_e$ a diagonal matrix having the element $d_{enn} = \sum_{m=1}^{M} a_{enm}$. Next the signs of $\mathbf{r}_{(n)}$ are randomized. This is equivalent to a phase randomization where phase changes are constrained to be $\pm\pi$ so that the synthesized instance keeps real. Then the inverse GFT is computed. All this, in conjunction with (3) can be compactly expressed for the computation of the synthetic instance $\mathbf{s}_{(n)}$

$$\mathbf{s}_{(n)} = \left(\mathbf{U}_e^H\right)^{-1} \mathbf{\Phi} \mathbf{U}_e^H \mathbf{x}_{(n)} = \mathbf{U}_e \mathbf{\Phi} \mathbf{U}_e^H \mathbf{x}_{(n)} \qquad , \qquad (4)$$

where we have used the fact that $\mathbf{U}_e$ is an unitary matrix (eigenvectors are orthonormal) because the Laplacian $\mathbf{L}_e$ is a real symmetric matrix. The matrix $\mathbf{\Phi}$ is a diagonal matrix having randomly selected values of 1 or -1, thus producing random sign changes in the transformed domain.

3.2 The discriminative block

Let us consider N original instances $\mathbf{x}_{(n)}$ $n=1\ldots N$ and a synthetic instance $\mathbf{s}_{(m)}$ computed from one of the original instances $\mathbf{x}_{(n)}$ using (4). The discriminator is to be designed so that it cannot be “cheated” by $\mathbf{s}_{(m)}$, i.e., we need a discriminant function that can discriminate between the class of the original instance $\mathbf{x}_{(m)}$ and the class of the synthetic instance $\mathbf{s}_{(m)}$. Considering the structured model of equation (1), the discriminant function can be applied on similarities between maximal cliques. Hence, let us define the vectors

$$\begin{aligned}\mathbf{k}^{x}_{(n)} &= \left[k\left(\mathbf{x}^{1}_{(m)},\mathbf{x}^{1}_{(n)}\right)\ldots k\left(\mathbf{x}^{C}_{(m)},\mathbf{x}^{C}_{(n)}\right)\right]^{T} \\ \mathbf{k}^{s}_{(n)} &= \left[k\left(\mathbf{s}^{1}_{(m)},\mathbf{x}^{1}_{(n)}\right)\ldots k\left(\mathbf{s}^{C}_{(m)},\mathbf{x}^{C}_{(n)}\right)\right]^{T}\end{aligned} \quad , \qquad (5)$$

where $k\left(\mathbf{x}^{c},\mathbf{y}^{c}\right)$ is a similarity measure (to be defined later) between the c-th maximal cliques of $\mathbf{x}$ and $\mathbf{y}$. Then $\mathbf{k}^{x}_{(n)}$ $n=1...N$ and $\mathbf{k}^{s}_{(n)}$ $n=1...N$ are feature vectors respectively corresponding to two different classes, and the discriminant is to be designed to separate both. The usual discriminator of a GAN scheme is a neural network (Goodfellow, 2016), although other classical alternatives for two-class problems exist like logistic regression (Menard, 2002) or the linear discriminant (Duda, Hart & Stork, 2000). Considering that we face a problem of extreme scarcity of original instances (very small N), we are restricted to using a discriminant function as simple as possible, so that it could be reasonably learned with the reduced amount of available data. Hence we propose the use of a linear discriminant. The discriminator can be a simple hard linear detector

$$\mathbf{k}^{T}\mathbf{w} \mathop{\gtrless}\limits^{H_1}_{H_{-1}} 0 \quad , \qquad (6)$$

where H_1 and H_{-1} respectively define the class or hypothesis corresponding to the original instance $\mathbf{x}_{(m)}$ and the synthetic instance $\mathbf{s}_{(m)}$. The optimum linear discriminant $\mathbf{W}$ can be obtained from the training set (5) as the solution to the linear system of equations

$$\underbrace{\mathbf{K}}_{(2N\times C)} \cdot \underbrace{\mathbf{w}}_{(C\times 1)} = \underbrace{\mathbf{v}}_{(2N\times 1)} \qquad \mathbf{K} = \left[\mathbf{k}^{x}_{(1)}\ldots\mathbf{k}^{x}_{(N)}\mathbf{k}^{s}_{(1)}\ldots\mathbf{k}^{s}_{(N)}\right]^{T} \qquad \mathbf{v} = \begin{bmatrix}\mathbf{1}_{N} \\ -\mathbf{1}_{N}\end{bmatrix} \quad . \qquad (7)$$

This is an overdetermined system which can be solved by using the Moore-Penrose left pseudoinverse

$$\mathbf{w}_{opt} = \mathbf{K}^{T}\left(\mathbf{K}\mathbf{K}^{T}\right)^{-1}\mathbf{v} \quad . \qquad (8)$$

3.3 The two blocks competition

Once we have separately described every block, let us put them together to present the complete algorithm of synthesis. The two blocks compete iteratively. Let us assume that the generative block generates and initial synthetic signal $\mathbf{s}_{(m,0)}$ obtained from one of the original instances $\mathbf{x}_{(m)}$ using (4). Then, the discriminative block computes the coefficients $\mathbf{w}_{opt,0}$ from (8) to optimally discriminate $\mathbf{s}_{(m,0)}$ from $\mathbf{x}_{(m)}$. Considering (6), we can verify the actual discrimination achieved by a raw estimate of the probability of error, which can be obtained dividing by $2N$, the number of times that $\mathbf{k}_{(n)}^{x\ T}\mathbf{w}_{opt,0} < 0$ and $\mathbf{k}_{(n)}^{s\ T}\mathbf{w}_{opt,0} > 0$. A value close to 0.5 of the estimated probability of error indicates that the discriminator is not able to distinguish the original from the synthetic instance. A value close to 0 indicates good discrimination. In this later case, we must try to correct the synthetic instance so that it can be accepted as an original instance. This is faced in the following.

Considering the form (equation (6)) in which the discriminator verifies if a given feature vector $\mathbf{k}$ belongs to H_1 or to H_{-1}, the generator may correct the vectors $\mathbf{k}_{(n)}^{s} \equiv \mathbf{k}_{(n,0)}^{s}$ $n = 1 \ldots N$ by multiplying every component by a correcting factor. Let us define the vector of correcting factors $\mathbf{f}_0 = \left[\, f_0^1 \cdots f_0^C \,\right]^T$. The corrected feature set is the dot product of every vector of the current set by $\mathbf{f}_0$, i.e.,

$$\mathbf{k}_{(n,1)}^{s} \equiv \mathbf{k}_{(n,0)}^{s} \cdot \mathbf{f}_0 = \left[k\left(\mathbf{s}_{(m,0)}^{1}, \mathbf{x}_{(n)}^{1}\right) \cdot f_0^1 \;\ldots\; k\left(\mathbf{s}_{(m,0)}^{C}, \mathbf{x}_{(n)}^{C}\right) \cdot f_0^C \right]^T \qquad n = 1 \ldots N \quad . \quad (9)$$

The goal of the correction is to "cheat" the discriminator, then the correcting vector can be obtained by the solution to

$$\underbrace{\left[\mathbf{k}_{(1,0)}^{s} \cdot \mathbf{f}_0 \;\ldots\; \mathbf{k}_{(N,0)}^{s} \cdot \mathbf{f}_0 \right]^T}_{(N\times C)} \cdot \underbrace{\mathbf{w}_{opt,0}}_{(C\times 1)} = \underbrace{\left[\, \mathbf{1}_N \,\right]}_{(N\times 1)} \Leftrightarrow \underbrace{\left[\mathbf{k}_{(1,0)}^{s} \cdot \mathbf{w}_{opt,0} \;\ldots\; \mathbf{k}_{(N,0)}^{s} \cdot \mathbf{w}_{opt,0} \right]^T}_{(N\times C)} \cdot \underbrace{\mathbf{f}_0}_{(C\times 1)} = \underbrace{\left[\, \mathbf{1}_N \,\right]}_{(N\times 1)}$$

$$. \quad (10)$$

This is again an overdetermined system which can be solved by using the Moore-Penrose left pseudoinverse

$$\mathbf{f}_{opt,0} = \mathbf{K}_{w,0}^{s}\left(\mathbf{K}_{w,0}^{s\ T}\mathbf{K}_{w,0}^{s}\right)^{-1}\left[\, \mathbf{1}_N \,\right] \qquad \mathbf{K}_{w,0}^{s} = \left[\mathbf{k}_{(1,0)}^{s} \cdot \mathbf{w}_{opt,0} \;\ldots\; \mathbf{k}_{(N,0)}^{s} \cdot \mathbf{w}_{opt,0} \right]^T \quad . \quad (11)$$

Notice that the optimum correcting factors correct the maximal cliques similarity measures, but ultimately we have to correct the initial synthetic signal cliques to get a new synthetic signal that can "cheat" the discriminator. Formally, we must find a correcting function for every maximal clique to get a new maximal clique $\mathbf{s}_{(m,1)}^{c} = g^c\left(\mathbf{s}_{(m,0)}^{c}\right)$ satisfying

$$k\left(\mathbf{s}_{(m,1)}^{c},\mathbf{x}_{(n)}^{c}\right)=k\left(\mathbf{s}_{(m,0)}^{c},\mathbf{x}_{(n)}^{c}\right)\cdot f_{opt,0}^{c} \qquad n=1...N \qquad . \tag{12}$$

The solution to this problem, if any, strongly depends on the specific similarity measure. Let us consider that we use the normalized correlation

$$k\left(\mathbf{s}_{(m,0)}^{c},\mathbf{x}_{(n)}^{c}\right)=\frac{\mathbf{s}_{(m,0)}^{c}{}^{T}\mathbf{x}_{(n)}^{c}}{\left\|\mathbf{s}_{(m,0)}^{c}\right\|\left\|\mathbf{x}_{(n)}^{c}\right\|} \qquad n=1...N \qquad , \tag{13}$$

where $\|\cdot\|$ stands for the Euclidean norm. It is not an easy matter to find the transformation $\mathbf{s}_{(m,1)}^{c}=g^{c}\left(\mathbf{s}_{(m,0)}^{c}\right)$ that, considering (13), complies with (12). However a change in the sign of $\mathbf{s}_{(m,0)}^{c}$ in (13) implies a change in the sign of $k\left(\mathbf{s}_{(m)}^{c},\mathbf{x}_{(n)}^{c}\right) \quad n=1...N$. Then we propose to use the transformation $\mathbf{s}_{(m,1)}^{c}=sign\left(f_{opt,0}^{c}\right)\mathbf{s}_{(m,0)}^{c}$ so that

$$\frac{\mathbf{s}_{(m,1)}^{c}{}^{T}\mathbf{x}_{(n)}^{c}}{\left\|\mathbf{s}_{(m,1)}^{c}\right\|\left\|\mathbf{x}_{(n)}^{c}\right\|}=k\left(\mathbf{s}_{(m,0)}^{c},\mathbf{x}_{(n)}^{c}\right)sign\left(f_{opt,0}^{c}\right)=k\left(\mathbf{s}_{(m,1)}^{c},\mathbf{x}_{(n)}^{c}\right) \qquad n=1...N \qquad . \tag{14}$$

This may be considered and approximation of (12) which only keeps the sign information of $f_{opt,0}^{c}$. This simplification admits a practical interpretation. Considering the hard decision (6) implemented by the discriminator, a positive value of $f_{opt,0}^{c}$ means that the current contribution of the feature $k\left(\mathbf{s}_{(m,0)}^{c},\mathbf{x}_{(n)}^{c}\right)$ favors the shift of the statistic towards H_1 (the discriminant is being cheated). On the contrary, a negative value of $f_{opt,0}^{c}$means that the current contribution of the feature $k\left(\mathbf{s}_{(m,0)}^{c},\mathbf{x}_{(n)}^{c}\right)$ favors the shift of the statistic towards H_{-1} (the discriminant is not being cheated). In both cases, the magnitude of $f_{opt,0}^{c}$ is a normalization adjustment to approximate the statistic $\mathbf{k}^{T}\mathbf{w}$ to 1 as much as possible, as imposed by the system of equations (10). In conclusion the sign of f_{0}^{c} indicates if the sign of the feature, and so the sign of the clique, must be changed or not to increase the difficulty of discriminating $\mathbf{s}_{(m)}$ from $\mathbf{x}_{(m)}$.

The above process describes the first iteration from $i=0$ to $i+1=1$of an iterative algorithm that can be repeated until the discrimination probability of error is close to 0.5. We will call it Generative Adversarial Network Synthesis for Oversampling (GANSO). A pseudocode description is given next, which uses GANSO to compute one synthetic instance from one original instance $\mathbf{x}_{(m)}$ selected from the set $\left\{\mathbf{x}_{(n)},\ n=1...N\right\}$of original instances.

Algorithm: GANSO

```
1: Input: Original instance set $\left\{\mathbf{x}_{(n)},\ n=1...N\right\}$, selected original instance $\mathbf{x}_{(m)} \in \left\{\mathbf{x}_{(n)},\ n=1...N\right\}$, adjacency matrix A, maximum number I of iterations for the generator/discriminator competitions.

# GENERATOR initialization
2: Initial synthetic instance eq. (4) $\mathbf{s}_{(m,0)} = \mathbf{U}_e \mathbf{\Phi} \mathbf{U}_e^H \mathbf{x}_{(m)}$
3: Segment $\mathbf{s}_{(m,0)} = \left[\mathbf{s}_{(m,0)1}^T \dots \mathbf{s}_{(m,0)L}^T\right]^T$ ; build cliques $\mathbf{s}_{(m,0)}^c = \left[\left(\mathbf{s}_{(m,0)1}^c\right)^T \dots \left(\mathbf{s}_{(m,0)L_c}^c\right)^T\right]^T$ $c=1...C$

# DISCRIMINATOR initialization
4: for $n=1...N$ do:
    5: Segment $\mathbf{x}_{(n)} = \left[\mathbf{x}_{(n)1}^T \dots \mathbf{x}_{(n)L}^T\right]^T$ ; build cliques $\mathbf{x}_{(n)}^c = \left[\left(\mathbf{x}_{(n)1}^c\right)^T \dots \left(\mathbf{x}_{(n)L_c}^c\right)^T\right]^T$ $c=1...C$
    6: Compute feature vector $\mathbf{k}_{(n)}^x$ as defined in (5) and (13)
7: end for

# BLOCKS competition
8: for $i = 0, 1 \dots I$ do:
    # DISCRIMINATOR
    9: for $n=1...N$ do:
        10: Compute feature vectors $\mathbf{k}_{(n,i)}^s$ as defined in (5) and (13)
    11: end for
    12: Compute optimum discriminator $\mathbf{w}_{opt,i}$ using (7) and (8)
    13: Compute probability of error as described after (8)
    14: If probability of error □0.5 then
        15: $\mathbf{s}_{(m)}^c = \mathbf{s}_{(m,i)}^c$ $c=1...C$ and go to 22
    16: else CONTINUE
    17: end if
    # GENERATOR
    18: Compute optimum correcting factors $\mathbf{f}_{opt,i}$ using (9)-(11)
    19: Compute corrected cliques $\mathbf{s}_{(m,i+1)}^c = sign\left(f_{opt,i}^c\right)\mathbf{s}_{(m,i)}^c$ $c=1...C$
20: end for

21: $\mathbf{s}_{(m)}^c = \mathbf{s}_{(m,I)}^c$ $c=1...C$
22: Built $\mathbf{s}_{(m)}$ from its maximal cliques
23: Output $\mathbf{s}_{(m)}$
```

4. Experiments

4.1 Simulated data

In this first experiment, we are going to illustrate the application of GANSO algorithm in a simulated scenario. This example has been selected due to its simplicity and to the connection with the real data application of the next section. We consider a two-class problem. The instances of both classes are vectors $\mathbf{x} \in \mathbb{R}^M$. The dimension *M* is restricted to be a multiple of 3, so that we may divide every instance into 3 segments $\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3$ of dimension *M*/3.

The assumed model for the instances is indicated in the following

$$\begin{array}{ll} \mathbf{x}_1 = a\mathbf{1}_{M/3} + \mathbf{e}_1 & \mathbf{e}_1 \sim N(\mathbf{0},\mathbf{I}) \\ \mathbf{x}_2 = b\mathbf{1}_{M/3} + \mathbf{e}_2 & \mathbf{e}_2 \sim N(\mathbf{0},\mathbf{I}) \\ \mathbf{x}_3 = a\mathbf{1}_{M/3} + \mathbf{e}_3 & \mathbf{e}_3 \sim N(\mathbf{0},\mathbf{I}) \end{array} \quad . \tag{15}$$

In Class 1 the values *a* and *b* are obtained by independently sampling uniform probability densities in the respective intervals (0, *A*) *A*>0 and (0, *B*) *B*>0 . In Class 2 the values *a* and *b* are obtained by independently sampling uniform probability densities in the respective intervals (0,-*A*) and (0,-*B*). Clearly, the presence of the same value *a* in segments 1 and 3 introduces some structural information in both classes. Let us put this more formally. We can compute the cross-correlation matrices between every pair of segments (notice that we assume that $a, b, \mathbf{e}_1, \mathbf{e}_2, \mathbf{e}_3$ are independent)

$$\begin{array}{l} E\left[\mathbf{x}_1\mathbf{x}_2^T\right] = E\left[a \cdot b\right]\mathbf{1}_{M/3}\mathbf{1}_{M/3}^T + \mathbf{1}_{M/3}E\left[a \cdot \mathbf{e}_2^T\right] + \mathbf{1}_{M/3}E\left[b \cdot \mathbf{e}_1^T\right] + E\left[\mathbf{e}_1\mathbf{e}_2^T\right] = \\ \qquad = E\left[a\right]E\left[b\right]\mathbf{1}_{M/3}\mathbf{1}_{M/3}^T = \frac{A}{2}\frac{B}{2}\mathbf{1}_{M/3}\mathbf{1}_{M/3}^T = E\left[\mathbf{x}_3\mathbf{x}_2^T\right] \\ E\left[\mathbf{x}_1\mathbf{x}_3^T\right] = E\left[a^2\right]\mathbf{1}_{M/3}\mathbf{1}_{M/3}^T = \frac{A^3}{3}\mathbf{1}_{M/3}\mathbf{1}_{M/3}^T \end{array} \quad . \tag{16}$$

Notice that $\mathbf{1}_{M/3}\mathbf{1}_{M/3}^T$ is an (*M*/3)x(*M*/3) all-ones matrix, so for *B*<<*A* the cross-correlation between segments 1-3 will be much greater than between segments 1-2, and 3-2 (actually it will be zero if *B*=0). We may assume in both classes a simple vMRF model of just three nodes, where node 1 is connected to node 3 and node 2 is left unconnected. Figure 2 depicts this simple graph.

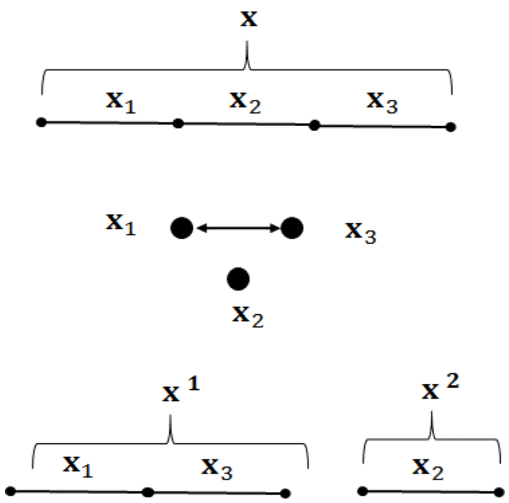

Figure 2. MRF and corresponding maximal cliques of the structure instance.

Therefore we have two maximal cliques. The first one formed by the union of the segments 1 and 3 and the second by just the segment 2. In the showed experiment we have selected the values *M*=21, *A*=3, *B*=0.3. Figure 3 shows superimposed 5 instances of Class 1 and another 5 of Class 2.

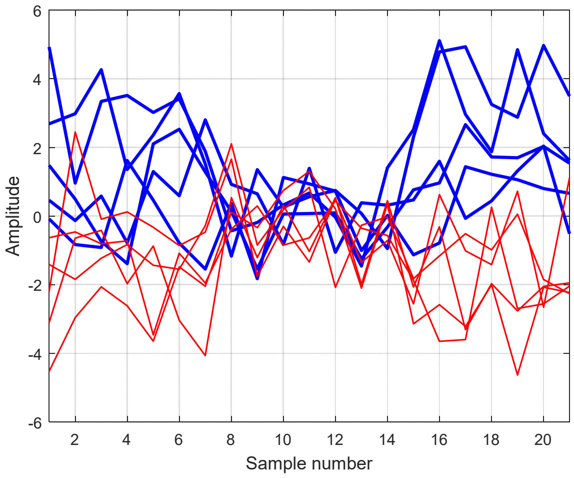

Figure 3. Five instances of Class 1 (blue) and five instances of Class 2 (red). *M*=21, *A*=3, *B*=0.3

Let us assume that we want to train a classifier to separate both classes, but we only have 5 instances of every class available. Obviously, we could use the simulation model (15) to synthesize as many instances as we want. However, to replicate a real data case with no model knowledge, we are going to use GANSO to increase the training set size. In the following we will use the term "original instance" to identify those instances generated by the simulation model (15)

First, let us show the relevance of the iterative competition between the two blocks of GANSO. We show in Figure 4 (left) 5 synthetic signals generated at iteration 0 of GANSO from, respectively, 5 original signals of the Class 1. We can see that lines number 4 (orange) and number 5 (magenta) appear with a modified polarity in segments 1 and 2. This is because the vMRF of Figure 2 is in force as far as segments 1 and 2 share the same polarity (positive or negative). However, after a few iterations, the polarity of these two segments is corrected (see Figure 4, right) while the rest of instances remain unchanged.

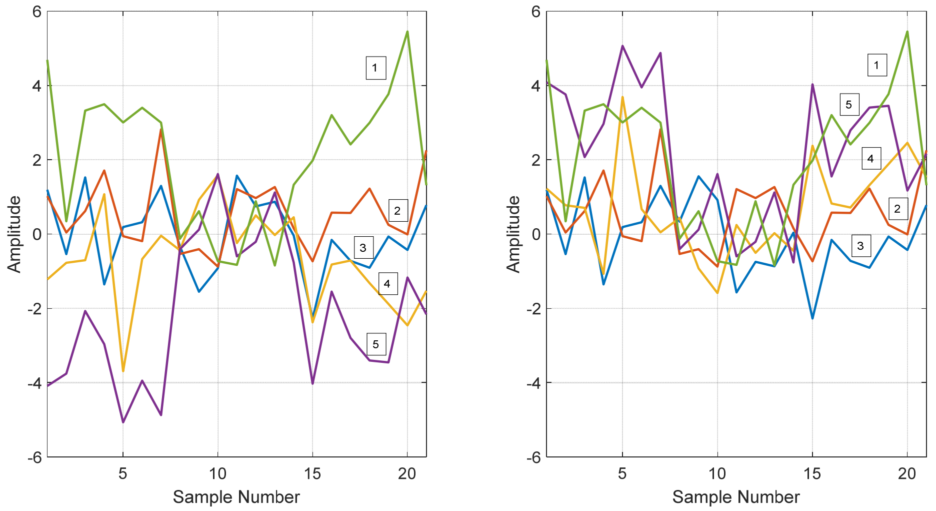

Figure 4. Five synthetic instances of Class 1 at iteration 0 of GANSO (left) and corrected instances after a few iterations of GANSO (right)

Finally, Figure 5 shows the learning curves corresponding to a linear discriminant which is to classify between Class 1 and Class 2. We show the probability of error for an increasing training set size per class varying from 10 to 80 in steps of 5 instances. Notice that, ultimately, we want to show the capability to reduce the classifier probability of error by adding synthetic instances to the training set. Thus we need an estimate of the probability of error as reliable as possible. In a real data application (see sections 4.2 to 4.4) we will be constrained by a reduced number of available original instances, not only for training but also for testing. So we will be forced to resort to random partitions of testing-training sets. However, in this simulation framework we can provide as many testing original instances as we want for a better estimation of the probability of error. Hence, the probability of error was estimated with a testing set of 100 original instances not including the 5 used for training. Three different types of training sets were considered to get the three different curves of Figure 5. The yellow one corresponds to training sets of only original instances (not included in the test set to avoid overfitting). As expected, this yields the best results. The blue line corresponds to a training set formed by 5 original instances per class plus 5*D* synthetic instances per class generated by GANSO, with *D* varying from 1 to 15, so that the total training set size still varies from 10 to 80. We can see that training with the GANSO synthetic instances gives a significant learning capability to the classifier, in spite of the using just 5 original instances for training. We observe that the learning curve has an initial fast descent, i.e., by adding some 10 synthetic instances (training set size of 15), the initial 0.50 probability of error of a random detector is reduced to just 0.28. From that point on, the learning curve decreases at a smaller rate. For example, notice that 65 synthetic instances added to the available 5 original instances (training set size of 70) yield a probability of error similar to the one achieved by training with some 30 original instances. In a real case scenario, this saving of 25 original instances can be very relevant. We have also included the results corresponding to the benchmark oversampling method SMOTE. We have essentially followed the standard procedure indicated in (Fernández et al., 2018). For every available original instance, we compute the difference between it and one of the other available original instances randomly selected. This difference is multiplied by a random number uniformly distributed between 0 and 1, and then, it is added to the original instance under consideration. This is repeated D times until we have the required number of $5D$ synthetic instances for every point in Figure 5. We can see in Figure 5 that SMOTE has no learning capability in this scenario of very small number of original data. As it was commented in the Introduction, recovering the original MPDF by interpolating a very small number of instances is not feasible. Other variants of SMOTE were tried, but the results were quite similar. This is again due to the assumed very small sample size scenario. Thus, for example, ADASYN generates more synthetic instances from those original instances having more neighbors of the majority class. In our case both classes has a very small number of original instances, i.e., there is not a majority class, hence that measure of vicinity is not very stable.

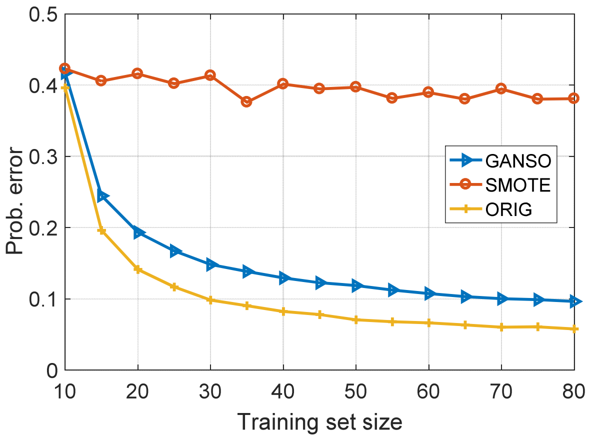

Figure 5. Learning curves of a linear classifier of classes 1 and 2 for different training sets: all original instances (yellow), 5 original+5D GANSO synthetic (blue) and 5 original+5D SMOTE synthetic (red). D varies from 1 to 15

4.2 Real data application 1

Neurological activity of patients can be assessed by means of specialized tests involving audio and/or visual stimuli. Thus, the so called “Barcelona test” (BT) (Quintana, 2010) encompasses a battery of tests designed in Spain in 1977 to evaluate higher mental functions. In this section, we present an experiment which implement an abbreviated subtest of the BT family: a visual short-term memory task. The subject is shown an item in the computer monitor screen for 10 seconds, and after a 10-second retention interval, he is asked to recognize the previously seen item among a set of four similar items. Once recognized, the subject press the keyboard and a new trial starts. A total of 10 trials are implemented having increasing difficulty. During the test, 18 bipolar EEG channels are recorded from the subject. Every channel is band-pass filtered between 0.5 and 30 Hz and sampled at a sampling frequency of 500 Hz. In this case, the objective is to verify to what extend the EEG signals might demonstrate changes in the neurological activity of the patient as he commutates from the stimuli phase to the retention+response phase. This information, in combination with the test results (number of correct answers) may help a better diagnosis of the subject neurological condition.

Then, we have implemented a two-class classifier, where Class 1 corresponds to the stimuli phase and Class 2 to the retention+response phase. From every EEG signal we have extracted 7 features in non-overlapped epochs of 0.25 seconds: sample mean, sample mean absolute value, centroid frequency, and powers in the delta (0.5-4 Hz), theta (4-8 Hz), alpha (8-13 Hz) and beta (13-30 Hz) frequency bands. From these features we form epoch instances of dimension 7. On the other hand, we know the initial and final instants of the two phases, so we can compute labelled phase instances by averaging all the epoch instances included inside the same phase interval. Thus, we obtain one labelled instance of every class for each trial up to a total of 10 labelled instances of Class 1 and 10 labelled instances of Class 2. This implies a very small number of labelled instances for both training and testing the classifier (for example 5 for training and 5 for testing). Certainly, we could increase the number of trials, but the subject will become progressively tired and the results will not be reliable. Then, this seems an appropriate scenario to experiment GANSO.

To this aim, we have considered the availability of 18 EEG channels. It has been demonstrated elsewhere (see for example Salazar, Safont & Vergara, 2019) that the different EEG channels exhibit different levels of correlation. So we may combine the 7-dimension phase instances of every single channel to form higher-dimension instances with some structural information. Obviously, there are many possible instance combination alternatives, but we have selected to combine just three channels to form instances of 21-dimension. Moreover, the channels are selected so that the first and the third have high correlation, while the intermediate is uncorrelated with the other 2. Thus, we reproduce a similar case to the one showed in the simulations of the previous section. Figure 6 shows the learning curves corresponding to a linear discriminant which is to classify between Class 1 and Class 2 for two different channel triads: 8-4-16 and 4-5-12. Six different subjects were separately considered. All of them suffered from neurological diseases yielding temporal lobe epileptic seizures of different degree. We show for every subject the probability of error for an increasing training set size per class varying from 10 to 80 in steps of 5 instances. The training set was formed by 5 original instances per class plus 5*D* synthetic instances per class, with *D* varying from 1 to 15. We have used both GANSO and SMOTE methods to generate synthetic instances. The testing set was formed by the remaining 5 original instances per class not selected for training. To get stable results, the showed probability of error is an average over the probability of error corresponding to 250 different 5+5 partitions of the 10 original instances per class.

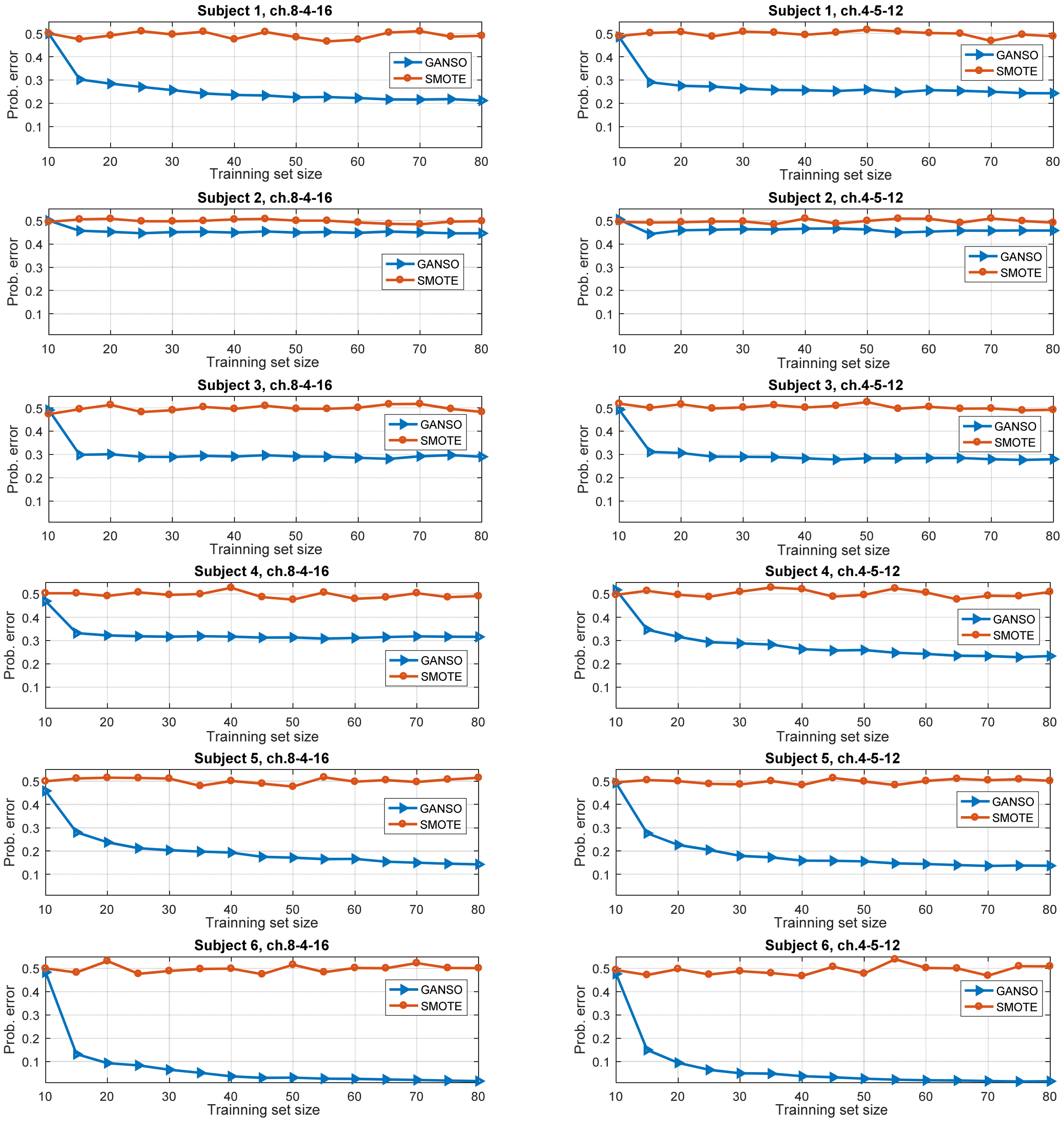

Figure 6. Learning curves of a linear classifier of classes 1 and 2 for different training sets: 5 original+5D GANSO synthetic (blue) and 5 original+5D SMOTE synthetic (red). D varies from 1 to 15. Six subjects are considered as well as two different channel triads, 8-4-16 (left), 4-5-12 (right)

We can see in Figure 6 that oversampling the training set by SMOTE cannot effectively improve the performance of the linear classifier in all subjects. However, GANSO provides synthetic training instances that achieve learning capability. Notice again the same fast descent of the learning curve from an initial probability of error of 0.5 (random detector) to a smaller value for 10 added synthetic instances (training set size of 15). That smaller value varies significantly among the different subjects. Notice that the classifier performance is indicative of the brain capability to commutate from one phase to the other during the implementation of the BT. Hence, the learning curve could be used as an additional element of diagnosis. Thus, for example, the information provided by the EEG signals of subject 2 seems to be very poorly related to the commutation between phases. On the other extreme, the EEG signals of subject 6 are clearly connected to the phase commutation. This was verified to be reasonably consistent with the different characteristics of every patient as well as with the results of other psychological tests. Thus, subject 6, significantly younger than the rest, was the one with the shortest historical record of neurological disease, and his measured levels of attention/concentration and immediate visual memory were the highest. However subject 2 presents a large history of disease, with the highest rate of epileptic crisis among all the subjects and a low level of attention/concentration.

4.3 Real data application 2

This application is related to helping neurological disease diagnosis from functional Magnetic Resonance Images (fMRI) of individuals with schizophrenia. Notice that an fMRI scanning session could last more than 30 minutes when the patient is performing specific testing tasks. This uses to be very stressing. Moreover, schizophrenic individuals require some medical stabilization before the session. Thus, in general, it is of practical interest to reduce as much as possible the number of fMRI sessions per patient. This is especially relevant in some cases like monitoring one patient over time, e.g., considering if recently captured fMRI images belongs or not to the same class than the old ones. It is also of interest in the evaluation of a reduced set of patients grouped by some similarity condition (eg., they are relatives). So, the essential objective is to evaluate if a classifier could be trained with a very small number of fMRI sessions.

In this experiment we will use fMRI images from an open access database (https://openneuro.org/datasets/ds000115/), authored by (Repovs & Bard, 2012). Each input corresponds to the fMRI scanning of a patient performing the *N-back working memory* task (Gazzaniga et al., 2014). Images were collected by measuring the Blood-Oxygen-Level-Dependent (BOLD) associated to changes in neuron blood flow due to brain activity. Scanning was made in three different axes in steps of 4 mm, this defines volume images of 4 x 4 x 4 mm^3 voxel size. We have considered the time interval corresponding to the *1-back working memory task*, where 137 images were captured, one every 2.5 s to a total time of 2.5x137=342,5 s= 5.7

min. A total of twelve patients have been selected from the database, six of them (control) are healthy, and the other six suffer from schizophrenia This provides sets of three patients per class for training and three for testing in a two-class problem: healthy (class 1), non-healthy (class 2). Figure 7 shows twelve fMRI axial slices respectively corresponding to the twelve selected patients. Notice that, except for some artifacts probably due to eye movement, there are not obvious differences among the images. Thus, training a classifier from such a small number of images is a challenge.

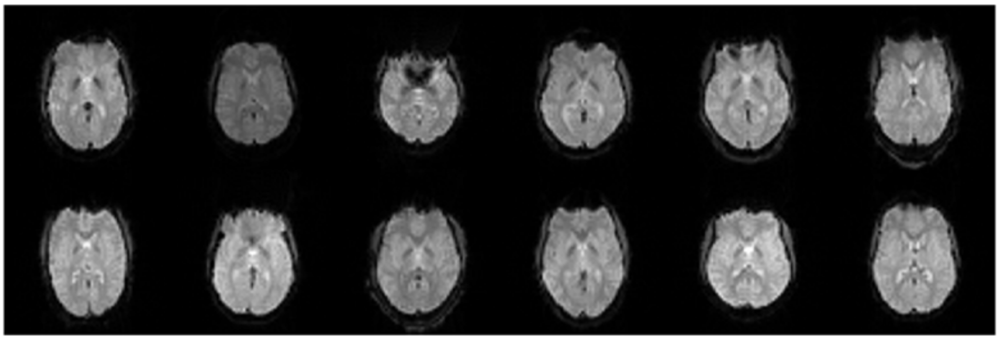

Figure 7. Axial views of fMRI slices from 12 subjects of the study measured at instant time 80 (200 s from the start of the *1-back working memory* task). The top row corresponds to individuals with schizophrenia (subject database identification 01, 05, 07, 09, 44, 60, from left to right) and the bottom row corresponds to healthy controls (subject database identification 11, 12, 15, 37, 46, 49, from left to right)

Before proceeding with the subsequent steps, every image must be preprocessed to compensate for some issues appearing during the acquisition process. Thus, we have implemented brain skull removal, slice time correction, motion compensation, and spatial smoothing (Lindquist, 2008; Jie et al., 2018).Then, we have applied the Automatic Anatomical Labelling (AAL) software package (Tzourio, 2002) to every preprocessed fMRI. AAL is typically used to build an atlas of 116 well established areas of the brain. Every area includes a given number of voxels. Let us consider the time series formed by the sequencing of the 137 BOLD amplitudes corresponding to the same voxel of the preprocessed fMRIs. We assign a unique time series to a given brain area by averaging the time series corresponding to all the voxels inside it. Thus, we finally have 116 BOLD time series corresponding to the 116 brain areas. On the other hand, brain connectivity has been recognized as a very relevant property related to neurological activity (Repovs & Bard, 2012; Salazar et al., 2019; Straathof et al., 2019; Mahjoory et al., 2017; Lang et al., 2012). Thus, we have estimated the connectivity between every two brain areas by computing the magnitude of the Pearson correlation coefficient (MPCC) between the corresponding pair of time series. Therefore, we obtained a matrix of 116 x116 MPCCs for every patient. Then, we computed the occurrence percentage (histograms) of MPCCs values in 15 uniform bins between 0 and 1. These 15 values are the features forming the instance to be used as input to the classifier. Figure 8 shows the instances thus computed corresponding to the six healthy patients (blue) and to the six non-healthy patients (red). We can see that instances are significantly overlapped, although the ones corresponding to healthy patients exhibit a small bias toward larger occurrences of high MPCCs. This is consistent with the expected loss of brain connectivity due to the presence of schizophrenia.

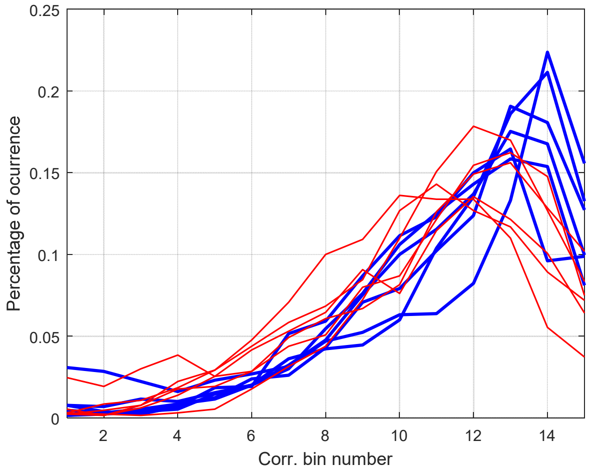

Figure 8. Instances of the six healthy patients (Class 1, blue) and the six non-healthy patients (Class 2, red)

The 15-dimension instances thus computed are considered to be divided into three 5-dimension non-overlapped segments so that we may assume the vMRF model of figure 2 to implement GANSO. This can be a reasonable assumption if we have a look to the instances of figure 8. Given that all the percentages of occurrence must add to 1, the first segment (bins 1 to 5) has lower amplitudes when the third segment (bins 11 to 15) has higher amplitudes and vice versa. This can be loosely considered a “negative correlation” between segments. On the other hand, the intermediate segment is left disconnected from the other two to allow some flexibility to GANSO in the generation of the synthetic instances. Notice that the definition of the structural constraints imposed by the vMRF do not need to be supported by strict mathematical models. It rather relates to some general properties of the original instances provided to the synthesizer to generate more appropriate instances. Figure 9 shows the learning curve corresponding to increasing sizes of the training sets. Every class is trained with only three out of six original instances plus a given number of synthetic instances. The testing set only includes the other three original instances not used for training. To get stable results, the showed probability of error is an average over the probability of error corresponding to 250 3+3 partitions of the 6 original instances per class. The starting point of the curves corresponds to training with just the three original instances of each class. At that point we get 0.47 probability of error, i.e., very close to a random detector. Then the training set size is increased by adding synthetic instances to the original three. We can appreciate an initial modest reduction of the probability of error for SMOTE. For example, by adding 27 synthetic instances (training set size of 30), the probability of error reduces to 0.41. However, GANSO gets a probability of error of 0.28 with the same number of 27 synthetic instances. From that point on, SMOTE cannot get further reductions, while GANSO slowly arrives to 0.25. This is additional evidence that GANSO provides learning capability even working with such a small number of available original instances. Notice the same effect observed in the previous experiments: an initial fast descent of the learning curve for a relative modest number of synthetic instances (some 9 instances for a total training set size of 12), followed by a slower descent.

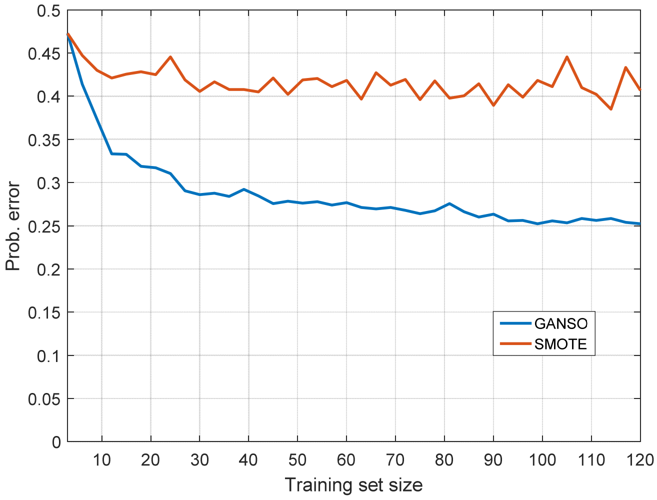

Figure 9. Learning curves of GANSO (blue) and SMOTE (red) of a linear classifier of classes 1 and 2 for different training set sizes. The first size is 3 (only original instances), the last size is 120 (3 original instances plus 117 synthetic instances)

4.4 Real data application 3

Determining the sleep stages of a patient through a long period of sleeping is significant to the diagnosis of sleep disorders. The sequential record of the different sleep stages is called the hypnogram (Jobert et al., 1994). Usually, hypnograms of a patient are manually determined by physicians from visual inspection of the so-called polysomnograms (PSG): a set of biosignals recorded from the patient while sleeping. This non-automatic procedure is tedious, long-lasting and sensitive to the physician fatigue: a typical sleeping period may last some 7 hours. Hence, automatic labelling of sleep stages is a very convenient option. Some automatic methods have been tested (Agarwal & Gotman, 2001; Safont et al., 2019) requiring large amounts of manually labelled stages to train the classifier. Thus, reducing the size of the training set as much as possible is of most practical interest. This clearly suggests a candidate application to be tested by GANSO.

The considered methods were applied on publicly available data from St Vincent's University Hospital/University College Dublin Sleep Apnea Database in Physionet (Heneghan, 2011; Goldberger et al., 2000). This database contains polysomnograms from 25 adult subjects (21 male, 4 female) with suspected sleep disorders. The database contains many kinds of physiological signals for every subject, as well as labeling for every 30-second epoch in sleep stages: wake, rapid eye movement (REM), and stages 1 (light sleep) to 4 (deep sleep). Of particular interest is the detection of wakefulness (arousals) (Jobert et al., 1994; Salazar, 2010) as they can be symptoms of apnea and epilepsy. So we have considered a two class problem. Class 1 corresponds to the wake stage and Class 2 to all the other stages. Moreover, to introduce a new signal modality with respect to the preceding applications, we have only considered the electrocardiogram (ECG) channel. ECG signal were sampled at 128 Hz and band-pass filtered between 0.3 and 75 Hz. The following 32 features were extracted from the ECG channel for each of the epochs: autoregressive model coefficients of order 4; Shannon's entropy maximal overlap discrete wavelet packet transform at level 4; multifractal wavelet leader estimates of the second cumulant of the scaling exponents and the range of Holder exponents leaders; and multiscale wavelet variance estimates up to fourth order using a Daubechies wavelet. These features are typically used in the literature of sleep staging (Li & Zhou, 2016). After extraction, the number of features was reduced by applying principal component analysis (PCA), resulting in12 features for an explained variance 98.9%. In this case, performing PCA has the additional interest of forming instances where we can assume the vMRF model of figure 2 to implement GANSO, given PCA provides features ordered in decreasing variance. Thus, the 12-dimension instances were considered to be divided into three 4-dimension non-overlapped segments. As the total explained variance is a constant number, increasing the amplitudes of the first segment implies a decreasing of the amplitudes of the third segment and viceversa. This rationale is quite similar to the previous one regarding the histogram features of figure 8. For every subject we have computed the original instances formed by the amplitudes of the 12 PCA features corresponding to the first 10 epochs manually labelled for Class 1 (wakes) and the first 10 epochs manually labelled for Class 2 (distributed in 2 epochs for every different non-wake stage: REM, stage 1, stage 2, sage 3 and stage 4). Figure 10 shows the 10 selected instances of Class 1 (blue) and Class 2 (red), for 4 representative subjects (profiles are given in Table 1). Notice the general amplitude decreasing consistent with the variance ordering of the PCA features.

Subject number	Gender	Age	PSG AHI (number of apneas and hypoapneas per hour)	Number of 30 s epochs required to manually label the first 10 stages of Class 1 and Class 2	Total number of 30 s epochs
5	M	47	12	79 (0,65 h)	813 (6,77 h)
16	M	49	16	79 (0,65 h)	852 (7,10 h)
18	F	41	13	85 (0,71 h)	913 (7,60 h)
23	M	49	14	80 (0,66 h)	838 (6,98 h)

Table 1. Profiles of 4 representative subjects

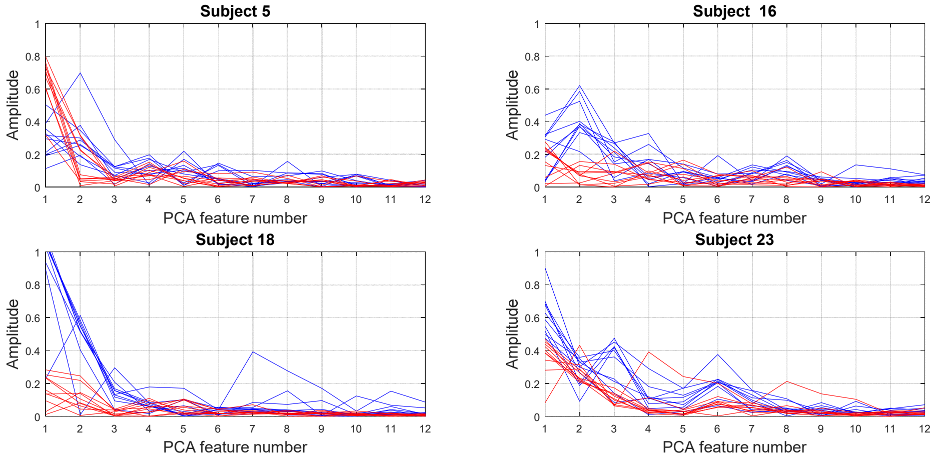

Figure 10. Selected instances of 12 PCA features corresponding to wake stages (Class 1, blue, 10 instances) and non-wake stages (Class 2, red, 10 instances), for four representative subjects

Then we computed the learning curves; the training set was formed by 5 of the original selected instances per class plus 5*D* synthetic instances per class, with *D* varying from 1 to 15. The other 5 original instances were used for testing. A total of 250 5+5 partitions of the 10 original instances per class were considered to compute average values. Figure 11 shows the learning curves corresponding to the four subjects of Figure 10.

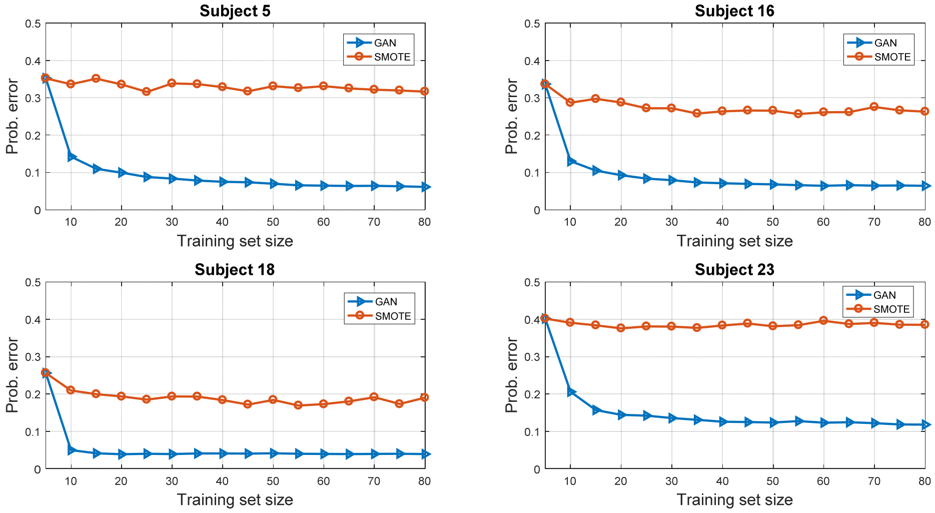

Figure 11. Learning curves of GANSO (blue) and SMOTE (red) of a linear classifier of classes 1 and 2 for different training set sizes. The first size is 5 (only original instances), the last size is 80 (5 original instances plus 75 synthetic instances)

We can draw similar conclusions to the ones corresponding to the previous real data experiments. GANSO yields a significant reduction of the initial probability of error by adding synthetic instances to the original ones, while SMOTE is not able to get that reduction. Notice again the initial fast descent of the learning curve for a relative modest number of synthetic instances followed by a slower descent. Similar learning curves were obtained for all the 25 subjects of the database.

Finally, a relevant data showed in Table 1 is the number of 30 s epochs required to manually label the first 10 stages of Class 1 and Class 2. The physician can start to label the 30 s epochs from the beginning of the ECG signal. Once there are at least 10 epochs from Class 1 and 10 from Class 2, manual labeling can stop. Then the automatic classifier is trained/tested from the 10+10 corresponding original instances as already explained, so that it can be applied to label the rest of the study. We can see in Table 1 that a saving above 90% of labeling the total sleep time is possible, thus largely relieving the manual labeling.

5. Conclusions

We have presented a new oversampling method, termed GANSO, to alleviate the limitations of classifier training in scenarios with extreme data scarcity. This later is compensated by some assumed knowledge about the inherent structure of the instances defined by a vMRF. This structural information is exploited by a GAN in both the discriminative and the generative blocks. Thus, the discriminative block implements a linear discriminant on features measuring the similarities between the cliques of the input instance and the corresponding cliques of the original labelled instances. On the other hand, the generative block extends the vMRF to the sample scale to synthesize surrogates by the Graph Fourier Transform. Both blocks iterate until the linear discriminant cannot distinguish the synthetic form the original instances. In the presented experiments, 1 to 5 iterations of GAN were enough for the acceptation of the synthetic instance.

We have demonstrated, both in simulated and real data experiments that GANSO is able to reduce the probability of error of the almost random detector corresponding to training with a very small training set size (only 3 or 5 original instances in the presented experiments). However, the benchmark method SMOTE cannot effectively improve the performance of a classifier with such a very small training set.

The applications considered were classification of stages of neuropsychological tests using EEG and FMRI data and classification of sleep stages using electrocardiographic (ECG) data for help in diagnosing of epilepsy, schizophrenia, and sleep disorders, respectively. We have observed in all the learning curves an initial fast decreasing of the probability of error. As a rule of thumb, this initial reduction is achieved by adding a number of synthetic signals equal to two or three times the available number of original signals. Further inclusion of more synthetic signals achieves a continuous reduction of the probability of error but at a much smaller decreasing rate. As a general conclusion, we may say that the vMRF structural information and the validity verification of GAN that GANSO incorporates, are crucial elements to generate valid synthetic instances to train the classifier.

Acknowledgments

This work was supported by Spanish Administration (Ministerio de Ciencia, Innovación y Universidades) and European Union (FEDER) under grant TEC2017-84743-P.

References

Agarwal, R., Gotman, J. (2001). Computer-assisted sleep staging. *IEEE Trans. On Biomedical Engineering, 48*, 1412-1423.

Angeletti, F., Bertin, E., & Abry, P. (2013). Random vector and time series definition and synthesis from matrix product representations: from statistical physics to hidden Markov models. *IEEE Transactions on Signal Process*ing, *61*, 5389-5400.

Belda, J., Vergara, L., Salazar, A., Safont, & G. Parcheta, Z. (2019). A new surrogating method by the Complex Graph Fourier Transform. *Entropy, 21-759* (2019), 1-18.

Beleites, C., Neugebauer, U., Bocklitz, T., Krafft, C. & Popp, J.: (2013). Sample size planning for classification models. *Analytica Chimica Acta*, *760*, 25–33.

Bhattacharyya, S., Jha, S., Tharakunnel, K., & Westland, J. C. (2011). Data mining for credit card fraud: A comparative study. *Decision Support Systems*, *50*, 602-613.

Borgnat, P., Abry, P., & Flandrin, P. (2012). Using surrogates and optimal transport for synthesis of stationary multivariate series with prescribed covariance function and non-Gaussian joint distribution. *In Proc. of IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)* (pp. 3729-3732), Kyoto, JAPAN.

Bunkhumpornpat, C., Sinapiromsaran, K., & Lursinsap, C. (2009). Safe-level-SMOTE: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem. *In Proceedings of the 13th Pacific-Asia Conference on Advances in Knowledge Discovery and Data Mining, (PAKDD '09)* (pp. 475–482), Bankok, THAILAND.

Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). SMOTE: Synthetic minority over–sampling technique. *Journal of Artificial Intelligent Research,16*, 321–357.

Chellappa, R., & Jain, A.K. (Eds) (1991). *Markov random fields, theory and applications*, Academic Press: Boston.

Duda, R. O., Hart, P. E. & Stork, D. H. (2000). *Pattern Classification (2nd ed.)*. Wiley Interscience: New York.

Fernández, A., Garcia, S., Herrera, F., & Chawla, N.V. (2018). SMOTE for learning from imbalanced data: progress and challenges, marking the 15-year anniversary. *Journal of Artificial. Intelligence Research*, *61*, 863-905.

Gazzaniga, M. S., Ivry, R. B., Mangun, & G. R. (2014). *Cognitive Neuroscience: The Biology of the Mind (4th Ed.).* W.W.Norton.

Goldberger, A., Amaral, L., Glass, L., Hausdorff, J., Ivanov, P. C., Mark, R. & Stanley, H. E. (2000). PhysioBank, PhysioToolkit, and PhysioNet: Components of a new research resource for complex physiologic signals. *Circulation*, *101*, e215–e220.

Goodfellow, I. (2016 December). Generative adversarial networks. Tutorial in *Neural Information and Processing System Conference*, Barcelona, SPAIN.

Guo, H., Li, Y., Gu, J., Huang, Y., & Gong, B. (2017). Learning from class-imbalanced data: review of methods and applications. *Expert Systems with Applications, 73*, 220-239.

Han, H., Wang, W. Y., & Mao, B. H. (2005). Borderline–SMOTE: A new over–sampling method in imbalanced data sets learning. *In Proceedings of the 2005 International Conference on Intelligent Computing (ICIC'05), Lecture Notes in Computer Science, 3644*, 878–887.

Han, X., Zhu, H., Yu, P., Wang, Z., Yao, Y., Liu, Z., & Sun, M. (2018). FewRel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. *In Conference on Empirical Methods in Natural Language Processing*, Brussels, BELGIUM.

He, H., Bai, Y., Garcia, E. A., & Li, S. (2008). ADASYN: Adaptive synthetic sampling approach for imbalanced learning. *In Proceedings of the 2008 IEEE International Joint Conference Neural Networks (IJCNN'08)*(pp. 1322–1328) , Hong Kong, CHINA.

He, H., García, E.A. (2009). Learning from imbalanced data. *IEEE Transactions on Knowledge and Data Engineering, 21*, 1263-1284.

Heneghan, C. (2001). St. Vincent's University Hospital / University College Dublin Sleep Apnea Database. Available: https://physionet.org/content/ucddb/1.0.0/.

Jie, B., Liu, M., Zhang, D., & Shen, D. (2018). Sub-network kernels for measuring similarity of brain connectivity networks in disease diagnosis. *IEEE Transactions on Image Processing*, *27*, 2340-2353.

Jobert, M., Shulz, H., Jähnig, P., Tismer, C., Bes, F. & Escola, H. (1994). A computerized method for detecting episodes of wakefulness during sleep based on the Alpha slow-wave index (ASI). *Sleep, 17*, 37-46.

Krawczyk, B. (2016). Learning from imbalanced data: open challenges and future directions. *Progress in Artificial Intelligence, 5*, 221-232.

Lake, B.M., Salakhutdinov, R., Gross, J., & Tenenbaum, J. (2011) One shot learning of simple visual concepts. *In Conference of the Cognitive Science Society (CogSci).*

Lafferty, J. D., McCallum, A., & Pereira, F. (2001) Conditional random fields: Probabilistic models for segmenting and labeling sequence data. *In 18th International Conference on Machine Learning*, Massachusetts, MA.

Lang, E.W., Tomé, A.M., Keck, I.R., Górriz-Sáez, J.M. & Puntonet, C.G. (2012). Brain connectivity analysis: a short survey. *Comput. Intell. Neurosci. 2012*, 1–21.

Li, R., Zhang, X., Chen, G., Mao, Y. & Wang, X. (2020). Multi-negative samples with Generative Adversarial Networks for image retrieval, Neurocomputing, 394, 146-157.

Li, T. & Zhou, M. (2016). ECG classification using wavelet packet entropy and random forests. *Entropy, 18*, *285*, 1-16.

Liao, T. W. (2008). Classification of weld flaws with imbalanced class data. *Expert Systems with Applications*, *35*, 1041–1052.

Lin, Z., Fanti, G., Khetan, A., & Sewoong, Oh. (2018). PacGAN: The power of two samples in generative adversarial networks. *Advances in Neural Information Processing Systems (NIPS 2018),31*, 1498-1507.

Lindquist, M.A. (2008). The statistical analysis of fMRI data. *Statistical Science, 23*, 439-464.

López, V., Fernández, A., García, S., Palade, V., & Herrera, F. (2013). An insight into classification with imbalanced data: empirical results and current trends on using data intrinsic characteristics. *Information Science, 250*, 113-141.

Mahjoory, K., Nikulin, V.V., Botrel, L., Linkenkaer-Hansen, K., Fato, M.M. & Haufe, S.(2017). Consistency of EEG source localization and connectivity estimates. *Neuroimage 152*, 590–601.

Mandic, D.P. , Chen, M., Gautama, T., Van Hulle, M.M., & Constantinides, A. (2008). On the characterization of the deterministic/stochastic and linear/nonlinear nature of time series. *In Proc. of the Royal Society A*, *464*, 1141–1160.

Menard, S. (2002). *Applied logistic regression analysis*, Sage Publications: London.

Miralles, R., Vergara, L., Salazar, A., & Igual, J. (2008). Blind Detection of Nonlinearities in Ultrasonic Grain Noise. *IEEE Trans. on Ultrasonics, Ferroelectrics, and Frequency Control*, *55*,637-647.

Pérez-Cruz, F., Pontil, M., & Ghahramani, Z. (2007). Conditional graphical models. In G. Bakir, T. Hofmann, B. Schölkopf, A.J.Semola, B. Taskar & S. V. N. Vishy Vishwanathan (Eds.), *Predicting Structured Data* (pp. 265–282). MIT Press.

Pirondini, E., Vybornova, A., Coscia, M., & Van De Ville, D. (2016). A Spectral Method for Generating Surrogate Graph Signals. *IEEE Signal Processing Letters*, *13*, 1275-1278.

Quintana, M. (2010). Spanish multicenter normative studies (Neuronorma project): norms for the abbreviated Barcelona Test. *Archives of Clinical Neuropsychology*, *26*, 144-157.

Repovs, G. & Bard, D.M. (2012) Working memory in healthy and schizophrenic individuals. *Frontiers in HUMAN NEUROSCIENCE, 6,* 1-15.

Salazar, A., Safont, G. & L. Vergara. (2018). Semi-supervised Learning for Imbalanced Classification of Credit Card Transactions. *In International Joint Conference on Neural networks (IJCNN 2018),* Rio de Janeiro, BRAZIL.

Salazar, A., Safont, G., & Vergara, L. (2019). A new graph based brain connectivity measure. *In International Work-conference in Artificial Neural Networks* (IWANN), Gran Canaria, SPAIN, 2019.

Salazar, A., Vergara, L. & Miralles, R. (2010). On including sequential dependence in ICA mixture models. *Signal Processing, 90*, 2314-2318.

Safont, G., Salazar, A., & Vergara, L. (2019). Multiclass Alpha Integration of Scores from Multiple Classifiers. *Neural Computation, 31*, 806-825.

Straathof, M., Sinke, M.R., Dijkhuizen, R.M. & Otte, W.M. (2019). A systematic review on the quantitative relationship between structural and functional network connectivity strength in mammalian brains. *J. Cereb. Blood Flow Metab. 39*, 189–209.

Su, L., Xu, X., Lu, Q., & Zhang, W. (2019). General image classification method based on semi-supervised generative adversarial networks. *High Technology Letters*, *25*, 35-41.

Tzourio, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer , B. & Joliot, M. (2002). "Automated Anatomical Labeling of activations in SPM using a Macroscopic Anatomical Parcellation of the MNI MRI single-subject brain". *NeuroImage*, 15, 273–289.

Vergara, L., Salazar, A., Belda, J., Safont, G., Moral & Iglesias, S. S. (2017). Signal Processing on Graphs for Improving Automatic Credit Card Fraud Detection. *International Carnahan Conference in Security Technology* (ICCST), Madrid SPAIN.