Classification of Multichannel Signals with Cumulant-based Kernels

Marco Signoretto, Emanuele Olivetti, Lieven De Lathauwer and Johan A. K. Suykens

***Abstract*—We consider the problem of training a discriminative classifier given a set of labelled multivariate time series (a.k.a. multichannel signals or vector processes). We propose a novel kernel function that exploits the spectral information of tensors of fourth order cross-cumulants associated to each multichannel signal. Contrary to existing approaches the arising procedure does not require an (often nontrivial) blind identification step. Nonetheless, insightful connections with the dynamics of the generating systems can be drawn under specific modeling assumptions. The method is illustrated on both a synthetic example as well as on a brain decoding task where the direction, either left of right, towards where the subject modulates attention is predicted from magnetoencephalography (MEG) signals. Kernel functions for unstructured data do not leverage the underlying dynamics of multichannel signals. A comparison with these kernels as well as with state-of-the-art approaches, including generative methods, shows the merits of the proposed technique.**

***Index Terms*—Multiple signal classification, Higher order statistics, Statistical learning, Kernel methods, Brain computer interfaces.**

I. Introduction

Classification of multivariate time series deviates from the standard pattern recognition framework where one deals with a set of measured attributes that do not change over time. A possible approach is based on the construction of metafeatures that exploit the additional structure inherited by the dependency over time [25] and then apply standard pattern recognition techniques on the derived set of attributes. An appealing class of tailored methodologies is based on probabilistic models, such as Hidden Markov models (HMMs), which have found successful applications in areas like speech, handwriting and gesture recognition.

In this paper we investigate a kernel-based discriminative methodology for the classification of multivariate signals. The idea makes use of higher order statistics to build a suitable similarity measure starting from the measured signals. Existing kernel-based approaches such as those in [2] and [46] are based on a two step procedure that first performs the blind identification of the generating systems (one for each multivariate time series) and successively carries out the classification by endowing the space of such models with a distance measure. In contrast, the approach that we propose here does not require postulating any specific model class nor the explicit identification of systems. In fact, the idea can be seen as entirely data driven. Nonetheless, as we show, if a certain structure can be assumed, interesting properties arise. By using higher order statistics we rely on the fact that the effect of transient, due to initial conditions, is negligible in the observed signals. In many practical applications the transient is discarded since it is believed to carry information potentially misleading for the analysis. For instance in the context of neuroscience, it is known that the encephalography (EG) signals immediately after the presentation of certain stimuli are non informative with respect to the mental processes of interest [45]. Likewise we assume that no marginally stable modes (corresponding to poles on the unit circle) are present and hence the steady-state output does not present persistent oscillations[1]. This assumption is usually met since in many practical applications signals are preprocessed to avoid persistent rhythmical components. In biosignal processing, for instance, these are often the effect of artifacts such as breathing and heart beating.

This paper is organized as follows. In Section II we formalize the problem of interest and discuss existing approaches for structured data. In Section III we recall instrumental facts about cumulant tensors of vector processes. In Section IV we introduce our novel cumulant-based kernel function and show properties under specific assumptions on the data generating mechanism. Section V is then concerned with the application that motivates this study, namely classification of (M)EG signals. In Section VI a comparison with alternative strategies is performed both on a synthetic as well as a real life example. We conclude by drawing our final remarks in Section VII.

II. Classification of Multichannel Signals

In the following we denote scalars by lower-case letters ($a, b, c, \ldots$), vectors as capitals ($A, B, C, \ldots$) and $N \times M$ matrices, namely elements of $\mathbb{R}^N \otimes \mathbb{R}^M$, as bold-face capitals ($\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C}, \ldots$). We also use lower-case letters i, j in the meaning of indices and with some abuse of notation we will use I, J to denote the index upper bounds. We write a_i to mean the $i-$th entry of a vector A and a_{ij} to mean $a_{ij} = (\boldsymbol{A})_{ij}$, the entry with row index i and column index j in a matrix $\boldsymbol{A}$. Finally we use $\mathbb{N}_I$ to denote the set $\{1, \ldots, I\}$.

Marco Signoretto and Johan A. K. Suykens are with the Katholieke Universiteit Leuven, Department of Electrical Engineering, ESAT, Kasteelpark Arenberg 10, B3001 Leuven (Heverlee), Belgium, tel:+32-(0)16320362, fax:+32-(0)16321970, (e-mail:{marco.signoretto,johan.suykens}@esat.kuleuven.be).

Emanuele Olivetti is with the NeuroInformatics Laboratory (NILab), Bruno Kessler Foundation and University of Trento, Trento, Italy (e-mail: olivetti@fbk.eu).

Lieven De Lathauwer is with the Group Science, Engineering and Technology of the Katholieke Universiteit Leuven Campus Kortrijk, E. Sabbelaan 53, 8500 Kortrijk, Belgium, tel: +32-(0)56326062, fax: +32-(0)56246999, (e-mail: lieven.delathauwer@kuleuven-kortrijk.be).

[1]This corresponds to having a purely stochastic vector process: no deterministic component is present in the Wold's decomposition of the process [47].

A. General Problem Statement

In this work we are concerned with classifying time-frames of vector processes. This task consists of correctly assigning a class label $y \in \mathscr{Y}$ to an input pattern

$$\boldsymbol{X} = [X(T_0),\ X(1+T_0),\ \cdots,\ X(T-1+T_0)] \in \mathbb{R}^D \otimes \mathbb{R}^T \tag{1}$$

that represents the evolution along successive time instants of a real $D-$variate discrete-time stationary vector process $\{X(t)\}$. This is accomplished by means of a *classifier*, namely a function

$$\bar{f} : \mathbb{R}^D \otimes \mathbb{R}^T \to \mathscr{Y} \tag{2}$$

inferred based upon a dataset of N input-target[2] pairs

$$\mathcal{D}_N = \left\{ \left(\boldsymbol{X}^{(n)}, y_n \right) \in \mathbb{R}^D \otimes \mathbb{R}^T \times \mathscr{Y} \ :\ n \in \mathbb{N}_N \right\} . \tag{3}$$

It is assumed that elements of $\mathcal{D}_N$ are drawn in a i.i.d. manner from the same unknown joint distribution $p(\boldsymbol{X}, y)$ that generates the test pair $(\boldsymbol{X}, y)$. Note that the marginal distribution $p(\boldsymbol{X})$ (as well as the conditional distribution $p(\boldsymbol{X}|y)$ on $\mathbb{R}^D \otimes \mathbb{R}^T$) is induced by the underlying mechanism generating $\{X(t)\}$.

B. Discriminative Versus Generative Methods

Consider for simplicity a binary classification problem for which[3] $\mathscr{Y} = \{-1, 1\}$. In order to build a classifier (2) one approach is to directly find a function

$$f : \mathbb{R}^D \otimes \mathbb{R}^T \to \mathbb{R} \tag{4}$$

so that $\bar{f} = \operatorname{sign} \circ f$ is as close as possible to some minimizer $\bar{h}$ of a suitable performance criterion such as the expected risk based on the $0-1$ *loss*:

$$\bar{h} \in \arg\min \left\{ E_{XY} \left[\delta(h(\boldsymbol{X}), y) \right] \ \text{s.t.}\ h : \mathbb{R}^D \otimes \mathbb{R}^T \to \mathscr{Y} \right\}$$

where δ denotes the Kroneker delta. Techniques following this strategy are called *discriminative methods* and include Support Vector Machines (SVMs). In contrast, *generative methods* such as Hidden Markov models (HMMs) first model the underlying probability distribution $p(\boldsymbol{X}, y)$ and then use the Bayes rule for determining the class membership. In this situation the classification rule (2) is therefore the result of a multi-step process. Discriminative methods have been shown to outperform generative methods in many applications; this applies in particular to the case where input data are naturally embedded in real vector spaces. The situation is less clear in those domains where input patterns possess additional structure, for instance inherited by the dynamical nature of the data, as in (1). In this case, in fact, it is generally difficult to extract features or metric relations to be used within discriminative approaches [24].

[2]As it will be manifested later an input pattern $\boldsymbol{X}$ should be considered as formed upon a time frame extracted from what corresponds to — in the standard state-space nomenclature — the output process X (compare equation (1) with equation (10)). Therefore we reserve the word *output* to refer to the output process in a state-space representation whereas we use *target* to address the target label.

[3]Multiclass classification problems ($\mathscr{Y}$ is a discrete set with more that two elements) are generally more involved but similar strategies apply.

C. Existing Kernels for Vector Processes

In the context of *kernel methods* [39] the difficulty of discriminative methods to deal with non-Euclidean data is overcome by defining suitable kernels [19] that serve as a similarity measure between pairs of input data. In particular, certain kernels that have been proposed for structured data can be adapted to the case where input patterns are time frames of vector processes, as in (1). In this case, (4) takes the form

$$f(\boldsymbol{X}) = \sum_{n \in \mathbb{N}_N} y_n \alpha_n k(q(\boldsymbol{X}^{(n)}), q(\boldsymbol{X})) + b \tag{5}$$

where, for any $n \in \mathbb{N}_N$, α_n weights the contribution of the $n-$th training data, $q : \mathbb{R}^D \otimes \mathbb{R}^T \to \mathscr{F}$ is a *processing step* that maps $\boldsymbol{X}$ into some metric space $\mathscr{F}$, k is a kernel function over $\mathscr{F}$ and finally b is a bias term. The approach proposed in [2], assumes that (1) is the output process of a linear non-Gaussian dynamical system. The processing step q corresponds to the explicit identification of a model as well as the recovery of the input process and the initial state associated to each output process. Remarkably, this deconvolution is a nontrivial step given that the generating systems might have non-minimal phase (as observed for the case of the human motion [2]). Finally k is a similarity measure over the parameter space of linear dynamical systems $\mathscr{F}$.

Related approaches are based on suitably defined instances of the *Fisher Kernel*. The Fisher Kernel represents a convenient way to incorporate generative models into discriminative classifiers [24], [44]. The key intuition behind it is that similar structured objects should induce similar log-likelihood gradients in the parameters of a predefined class of generative models [24]. Notably, as in [2], the use of Fisher kernels requires explicit estimation within a user-defined generative class.

Remark II.1. *In applications the choice of the generative class is often guided by computational tractability more than by principled insights. A sensible choice for multivariate time series is to assume* linear Gaussian models *[35]. For this case, practical expectation maximization (EM) algorithms exist for the estimation of the model parameters [20], [35]. A derivation of a Fisher kernel within this framework is concisely described in Appendix A for later comparisons.*

D. Main Contribution

In this paper we study a novel similarity measure for vector processes that, contrary to the aforementioned alternatives, does not require to assume a specific generative class nor to explicitly identify some model's parameter. In fact, the idea is based on the sample version of higher order statistics and therefore grounds solely on the stationarity and the ergodicity of the multichannel signals. More precisely, the proposed kernel makes use of *cumulant tensors*. These are multidimensional arrays whose entries correspond to cross-cumulants associated to the channels of a vector process. We exploit the algebraic structure of these objects by adapting to the present setting a family of kernel functions defined on tensors in [37]. Many types of structured data admit a natural representation as tensors; the family of tensorial kernels found

in [37] was proposed to deal with this general class of data; in the present paper we focus on dynamical systems and propose to use cumulant tensors within the tensorial kernel framework.

Although the applicability of the proposed kernel does not require postulating any specific model class, insightful properties can be derived as soon as certain generating models are assumed. In particular, we draw the link between the proposed similarity measure and the latent state process in the case of multi-input multi-output (MIMO) linear and time-invariant state-space models. This provides useful insights on certain processes of interest as for the case of EG signals that we consider.

III. Tensors and Cumulants

Next we bring in the instrumental facts on tensors and cumulants that we need for our novel cumulant-based kernel.

A. Tensors

We recall that $P-$th order tensors, which we denote by calligraphic letters ($\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, ...), are higher order generalizations of vectors (first order tensors) and matrices (second order tensors). Scalars can be seen as tensors of order zero. We write $a_{i_1,\ldots,i_M}$ to denote $(\mathcal{A})_{i_1,\ldots,i_M}$. The linear span of tensors form a vector space which we denote by $\mathbb{R}^{I_1}\otimes\mathbb{R}^{I_2}\otimes\cdots\otimes\mathbb{R}^{I_M}$. A *$m-$mode vector* of $\mathcal{A}\in\mathbb{R}^{I_1}\otimes\mathbb{R}^{I_2}\otimes\cdots\otimes\mathbb{R}^{I_M}$ is an element of $\mathbb{R}^{I_m}$ obtained from $\mathcal{A}$ by varying the index i_m and keeping the other indices fixed. The following definition is instrumental for introducing the cumulant-based kernel.

Definition III.1 ($m-$mode matricization)**.** *Let $J_m = I_1I_2\cdots I_{m-1}I_{m+1}\cdots I_M$. A $m-$th mode matrix unfolding of $\mathcal{A}\in\mathbb{R}^{I_1}\otimes\mathbb{R}^{I_2}\otimes\cdots\otimes\mathbb{R}^{I_M}$, denoted as $\mathcal{A}_{\langle m\rangle}$, is that $I_m\times J_m$ matrix whose columns correspond to all the $n-$mode vectors in any prescribed and fixed order.*

The next definition will be used to study properties of the cumulant-based kernel.

Definition III.2 ($m-$mode product)**.** *Let $J_m = I_1I_2\cdots I_{m-1}I_{m+1}\cdots I_M$. The $m-$mode product of a tensor $\mathcal{A}\in\mathbb{R}^{I_1}\otimes\mathbb{R}^{I_2}\otimes\cdots\otimes\mathbb{R}^{I_M}$ by a matrix $\boldsymbol{U}\in\mathbb{R}^{J_m\times I_m}$, denoted by $\mathcal{A}\times_m\boldsymbol{U}$, is that element of $\mathbb{R}^I_1\otimes\mathbb{R}^I_2\otimes\cdots\otimes\mathbb{R}^{I_{m-1}}\otimes\mathbb{R}^{J_m}\times\mathbb{R}^{I_{m+1}}\otimes\cdots\otimes\mathbb{R}^{I_M}$ defined by*

$$\mathcal{Y}=\mathcal{A}\times_m\boldsymbol{U}\ \Leftrightarrow\ \mathcal{Y}_{\langle m\rangle}=\boldsymbol{U}\mathcal{A}_{\langle m\rangle}\ .$$

B. Cumulants of Vector Processes

Cumulants are higher order statistics [31], [30] that so far have received interest especially for the identification of linear and non-linear systems [16], [21] as well as for blind source separation [8]. For a real-valued discrete-time and zero-mean stationary scalar process $\{x(t)\}$ [23], the $J-$th order cumulant function $c^x:\mathbb{Z}^{J-1}\to\mathbb{R}$ is defined in terms of the so called *cumulant generating function* [33]. For a real $D-$variate discrete-time stationary vector process[4] $\{X(t)\}$ one needs the notion of *cross-cumulants* between scalar entries of the vector. Suppose $E[X(t)]=0$. Then the second order cross-cumulant between the d_1-th and d_2-th entries of the vector process X is defined as:

$$c^X_{d_1d_2}(l_1):=E\left[x_{d_1}(t)x_{d_2}(t+l_1)\right]\ . \tag{6}$$

Higher order cross-cumulants are defined in a similar fashion. In particular, simple formulae exist up to the fourth order, see Appendix B.

[4]Sometimes we simply indicate it as X.

C. Cumulant Tensors

We now bring in the tensor representations of cumulants that we consider in this paper. To begin with, for a given finite $J\in\mathbb{N}$ and any $i\in\mathbb{N}_{J-1}$, let $\mathscr{L}_i\subset\mathbb{Z}$ be an arbitrary finite set with cardinality S^i and denote by l^i_s its $s-$th element. Consider now the collection of $\prod_{i\in J-1}S^i$ integer tuples obtained as

$$\mathscr{L}:=\mathscr{L}_1\times\mathscr{L}_2\times\cdots\times\mathscr{L}_{J-1}\ . \tag{7}$$

Definition III.3 ($\mathscr{L}-$Cumulant Tensor)**.** *We call $\mathscr{L}-$cumulant tensor, denoted as $\mathcal{C}^X_{\mathscr{L}}$, the $(2J-1)-$th order tensor*

$$\mathcal{C}^X_{\mathscr{L}}\in\underbrace{\mathbb{R}^D\otimes\mathbb{R}^D\otimes\cdots\otimes\mathbb{R}^D}_{J\text{ times}}\otimes\mathbb{R}^{S^1}\otimes\mathbb{R}^{S^2}\otimes\cdots\otimes\mathbb{R}^{S^{J-1}}$$

defined entry-wise by

$$\left(\mathcal{C}^X_{\mathscr{L}}\right)_{d_1d_2\cdots d_Js_1s_2\cdots s_{J-1}}:=c^X_{d_1d_2\cdots d_J}(l^1_{s_1},l^2_{s_2},\ldots,l^{J-1}_{s_{J-1}}) \tag{8}$$

for any tuple $[d_1,\ d_2,\ \cdots,\ d_J,\ s_1,\ s_2,\cdots,s_{J-1}]\in\mathbb{N}_D\times\mathbb{N}_D\cdots\times\mathbb{N}_D\times\mathbb{N}_{S^1}\times\mathbb{N}_{S^2}\times\cdots\times\mathbb{N}_{S^{J-1}}$.

Example III.4. *As a concrete example suppose that X is a zero-mean $4-$variate ($D=4$) stationary process. Take $J=3$ and let $\mathscr{L}_1=\{-3,1\}$ and $\mathscr{L}_2=\{-3,0,2\}$. In this case $\mathscr{L}=\{(-3,-3),(-3,0),\ (-3,2),\ (1,-3),(1,0),(1,2)\}$, and each entry of the cumulant tensor $\mathcal{C}_{\mathscr{L}}$, indexed by a tuple $[d_1,\ d_2,\ d_3,\ s_1,s_2]\in\mathbb{N}_4\times\mathbb{N}_4\times\mathbb{N}_4\times\mathbb{N}_2\times\mathbb{N}_3$, is a 3rd order cross-cumulant between the d_1-th, d_2-th and d_3-th entries of the vector process X.*

A special case that arises from the previous definition corresponds to the situation where for any $i\in\mathbb{N}_{J-1}$, the set $\mathscr{L}_i$ is a singleton. In this case $\mathscr{L}$ is also a singleton, i.e., we have $\mathscr{L}=\left\{[l^1_1,\ l^2_1,\ \ldots,\ l^{J-1}_1]\right\}$ for a single fixed tuple of integers $[l^1_1,\ l^2_1,\ \ldots,\ l^{J-1}_1]$. Additionally $J-1$ dimensions of $\mathcal{C}^X_{\mathscr{L}}$ are degenerate, that is, $\mathcal{C}^X_{\mathscr{L}}$ can be equivalently represented as a $J-$th order tensor. This is formalized in the next Definition.

Definition III.5 ($(l_1,\ l_2,\ \ldots,\ l_{J-1})$-Cumulant Tensor)**.** *We call $(l_1,\ l_2,\ \ldots,\ l_{J-1})-$cumulant tensor, denoted as $\mathcal{C}^X(l_1,l_2,\ldots,l_{J-1})$, the $J-$th order tensor*

$$\mathcal{C}^X(l_1,l_2,\ldots,l_{J-1})\in\underbrace{\mathbb{R}^D\otimes\mathbb{R}^D\otimes\cdots\otimes\mathbb{R}^D}_{J\text{ times}}$$

defined entry-wise by

$$\left(\mathcal{C}^X(l_1,l_2,\ldots,l_{J-1})\right)_{d_1d_2\cdots d_J}:=c^X_{d_1d_2\cdots d_J}(l_1,l_2,\ldots,l_{J-1})\ . \tag{9}$$

In the remainder of this paper we simply write *cumulant tensor* when the specific notion that we are referring to is clear from the context.

D. Sample versions

We write, respectively, $\hat{\mathcal{C}}^X_{\mathscr{L}}$ and $\hat{\mathcal{C}}^X(l_1, l_2, \ldots, l_{J-1})$ to denote the empirical (a.k.a. sample) versions of the cumulant tensors introduced in the previous definitions. These are simply obtained by replacing the expectation operator by the *time average* in the definitions of cross-cumulants (see equation (6) and Appendix B) [18], [42]. For instance, suppose that $\{X(1), X(2), \cdots, X(T)\}$ is a given sample from the process X. Then if we let $T_0 := \max\{1, 1 - l_1\}$ and $T' := \min\{T, T - l_1\}$, the empirical version of (6) is obtained as:

$$\hat{c}^X_{d_1 d_2}(l_1) := \frac{1}{T' - T_0 + 1} \sum_{t=T_0}^{T'} x_{d_1}(t) x_{d_2}(t + l_1) \ .$$

We also assume (as done everywhere else, at least implicitly) that sample versions of cumulants converge towards their theoretical counterpart for $T \to \infty$. This is true, in particular, whenever X is *strictly stationary* and *ergodic*[5] and $E[X_d^2(t)] < \infty$ for all $d \in \mathbb{N}_D$ [23, Section IV.2, Theorem 2]. However, as known, the property of ergodicity is not observationally verifiable from a single realization of a time series [23].

We now link a class of time series with their generating dynamical system. This will allow us to derive useful properties of the cumulant-based similarity measure that we introduce in the next Section.

E. Cumulants of Processes with State-Space Representation

Suppose that, for the given stationary process X, the *z-spectrum*[6] $S(z) := \sum_{l \in \mathbb{Z}} \mathcal{C}^X(l) z^{-l}$ is a rational function and has maximal normal rank everywhere on the complex unit circle. In this case a fundamental result [26] shows that there are $\boldsymbol{A}, \boldsymbol{B} \in \mathbb{R}^{D_Z} \otimes \mathbb{R}^{D_Z}, \boldsymbol{C} \in \mathbb{R}^{D} \otimes \mathbb{R}^{D_Z}$ and white processes[7] W (input process) and V (noise process) such that X can be interpreted as the output process of the MIMO stable system $\Sigma = (\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{C})$ in state-space form:

$$Z(t+1) = \boldsymbol{A} Z(t) + \boldsymbol{B} W(t) \quad (10)$$

$$X(t) = \boldsymbol{C} Z(t) + V(t) \ . \quad (11)$$

Conversely, starting from the generating dynamical system (10)-(11) we have the following basic result.

Proposition III.1 (State-space Model: Cumulant of the Output Process)**.** *Assume a stable state-space model* (10)-(11) *specified as above and suppose additionally that the processes W and V are mutually independent. For any tuple $[l_1,\ l_2,\ \ldots,\ l_{J-1}] \in \mathbb{Z}^{J-1}$ we have, in steady state*

1)

$$\mathcal{C}^X(l_1, l_2, \ldots, l_{J-1}) = \mathcal{C}^Z(l_1, l_2, \ldots, l_{J-1}) \times_1 \boldsymbol{C} \times_2 \boldsymbol{C} \times_3 \cdots \times_{J-1} \boldsymbol{C} + \mathcal{C}^V(l_1, l_2, \ldots, l_{J-1}) \ . \quad (12)$$

2)

If the noise process V is Gaussian and $J > 2$ then $\mathcal{C}^V(l_1, l_2, \ldots, l_{J-1})$ vanishes in (12)*.*

Proof: In steady state the state vector results from a linear filtering of the input process W and hence is statistically independent of the noise process V. Point 1 follows immediately from basic properties of cumulants, namely multilinearity and their additivity under the independence assumption (see e.g. [14, Sec. 2.2 properties 2 and 5]). Point 2 is a well known property of cumulants of a white Gaussian process [31]. ■

Results in line with Proposition III.1 can be found elsewhere. In particular, cumulants of the output process of a stable single-input single-output linear time-invariant system were studied in [5]. Cumulants of a broader class of linear MIMO systems (time-varing with possibly non-stationary inputs) were the subject of [43]. It is worth recalling that if a real random variable has symmetric distribution about the origin, then all its odd cumulants vanish [31]. In many cases this last condition is naturally verified for the entries of the state vector; in light of this fact and of point 2 of Proposition III.1 one then considers $J \geq 4$ in applications.

Generally speaking, it becomes harder to estimate cumulants from sample data as the order increases, i.e. longer data sets are required to obtain the same accuracy [3], [18], [42]. In the following we stick to $J = 4$, also used in the context of system identification [28]. This choice ensures that the output cumulant is independent of the noise process.

IV. Cumulant-based Kernel Approach for Multichannel Signals

The tensorial kernel [37] was originally proposed to exploit the spectral content of general tensors. Here we briefly recall its definition and then bring in cumulant tensors in order to establish a similarity measure between multichannel signals. We then show properties of the resulting cumulant-based kernel function.

A. General Factor Kernels and Tensorial Kernel

Let $\mathcal{X}_{\langle m \rangle}$ and $\mathcal{Y}_{\langle m \rangle}$ denote the $m-$th matrix unfolding of two generic elements $\mathcal{X}$ and $\mathcal{Y}$ of the same vector space $\mathbb{R}^{I_1} \otimes \mathbb{R}^{I_2} \otimes \cdots \otimes \mathbb{R}^{I_M}$. The singular value decomposition (SVD) of $\mathcal{X}_{\langle m \rangle}$ can be factorized in a block-partitioned fashion as follows:

$$\mathcal{X}_{\langle m \rangle} = \begin{pmatrix} \boldsymbol{U}^{(m)}_{\mathcal{X},1} & \boldsymbol{U}^{(m)}_{\mathcal{X},2} \end{pmatrix} \begin{pmatrix} \boldsymbol{S}^{(m)}_{\mathcal{X},1} & 0 \\ 0 & 0 \end{pmatrix} \begin{pmatrix} \boldsymbol{V}^{(m)\top}_{\mathcal{X},1} \\ \boldsymbol{V}^{(m)\top}_{\mathcal{X},2} \end{pmatrix} \quad (13)$$

where 0's are matrices of zeros with appropriate sizes. A similar decomposition holds for $\mathcal{Y}_{\langle m \rangle}$. Let now $g(\boldsymbol{V}^{(m)}_{\mathcal{X},1}, \boldsymbol{V}^{(m)}_{\mathcal{Y},1})$

[5] Different is the situation of signals driven by *cyclostationary* sources that we do not consider here. In this situation, which arise for instance in the context of radiocommunications, it is known that the empirical estimators arising from time averages are *biased*. See, e.g., [17] and [8, Section 17.2.2.2].

[6] Notice that $\mathcal{C}^X(l)$, corresponds to $E[X(t)X(t+l)^\top]$.

[7] We recall that a stationary stochastic process $X(t)$ is *white* if $E[X(i)X^\top(j)] = \boldsymbol{Q}\delta_{ij}$ where δ is the Kronecker delta function and $\boldsymbol{Q}$ denotes some covariance matrix.

be the distance function given by:

$$g\left(\boldsymbol{V}_{\mathcal{X},1}^{(m)},\boldsymbol{V}_{\mathcal{Y},1}^{(m)}\right):=\left\|\boldsymbol{V}_{\mathcal{X},1}^{(m)}\boldsymbol{V}_{\mathcal{X},1}^{(m)\top}-\boldsymbol{V}_{\mathcal{Y},1}^{(m)}\boldsymbol{V}_{\mathcal{Y},1}^{(m)\top}\right\|_F^2 . \tag{14}$$

The latter is defined in terms of the subspaces spanned by the right singular vectors corresponding to non-vanishing singular values and is known as *projection Frobenius-norm* [15], [37]. It can be restated in terms of *principal angles* that, in turn, are commonly used to define a metric over linear dynamical systems [11], [29]. We can now introduce the $m-$th *factor kernel* according to $k^m(\mathcal{X},\mathcal{Y}):=\exp(-\frac{1}{2\sigma^2}g(\boldsymbol{V}_{\mathcal{X},1}^{(m)},\boldsymbol{V}_{\mathcal{Y},1}^{(m)}))$ where σ is a user-defined tuning parameter. Finally we construct the positive definite *tensorial kernel function* k as the pointwise product of the factor kernels [37]

$$k(\mathcal{X},\mathcal{Y}):=\prod_{m\in\mathbb{N}_M}k^m(\mathcal{X},\mathcal{Y}) . \tag{15}$$

The latter is a positive type function that measures the similarity between two tensors based upon the spectral content of their matrix unfoldings, see [37] for details. Note that the induced similarity measure can be characterized from an algebraic perspective starting from the subspaces associated to the matrix unfoldings [12].

B. Cumulant-based Kernel Functions

We consider now a specific type of tensors, i.e., $\mathscr{L}-$cumulant tensors (see Section III-C). Notice that, since we stick to the case of fourth order cumulants, these tensors have in general order 7. One has $\mathscr{L}:=\mathscr{L}_1\times\mathscr{L}_2\times\mathscr{L}_3$ where, for any $i\in\{1,2,3\}$, $\mathscr{L}_i$ is an arbitrary finite subset of $\mathbb{Z}$. Given (15) it is now straightforward to introduce a cumulant-based kernel $k_{\mathscr{L}}:\mathbb{R}^D\otimes\mathbb{R}^T\times\mathbb{R}^D\otimes\mathbb{R}^T\to\mathbb{R}$ according to:

$$k_{\mathscr{L}}(\boldsymbol{X}_1,\boldsymbol{X}_2):=k\left(\hat{\mathcal{C}}_{\mathscr{L}}^{X_1},\hat{\mathcal{C}}_{\mathscr{L}}^{X_2}\right)=\prod_{p\in\mathbb{N}_P}k^p(\hat{\mathcal{C}}_{\mathscr{L}}^{X_1},\hat{\mathcal{C}}_{\mathscr{L}}^{X_2}) \tag{16}$$

where the empirical versions of the cumulants are computed based upon the observed time frames $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$.

Remark IV.1. *In* (5) *we introduced the general kernel-based model with q playing the role of a processing step. We note here that in* (16) *the processing step q corresponds to the evaluation of empirical expectations (see Subsection III-D). Contrary to the kernels for structured data discussed in Subsection II-C, no additional estimation steps are required.*

Remark IV.2. *A careful look at equations* (8) *and* (9) *reveals that entries of cumulant tensors depend only on time lags and channel indices. Therefore, since we consider stationary and ergodic signals,* (16) *does not require that $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ are aligned in time nor that the two sequences have the same length.* [8]

C. Computational Aspects

The freedom in constructing $\mathscr{L}$ gives rise to a whole family of cumulant-based kernel functions. For general $\mathscr{L}$, however, the evaluation of $k_{\mathscr{L}}$ at $\boldsymbol{X}_1$ and $\boldsymbol{X}_2$ requires computing the SVD of 14 matrices (7 for both arguments). The sizes of these matrices depend upon D and the cardinalities of $\mathscr{L}_i, i\in\mathbb{N}_3$. Since in many practical applications D is of the order of tens or even hundreds, $k_{\mathscr{L}}$ might be prohibitive to evaluate. In this respect we make the following observations:

- The case where $\mathscr{L}$ is singleton (Definition III.5) implies a significant reduction in the order of the tensor: in this case $\mathcal{C}_{\mathscr{L}}^{X_1}$ and $\mathcal{C}_{\mathscr{L}}^{X_2}$ can be equivalently represented as 4-th order tensors. For this reason in applications we consider $\mathcal{C}^X(0,0,0)$ namely, we take only the zero-lags tuple ($\mathscr{L}=\{[0,0,0]\}$)[9]. Note that in this case all the tensor matricizations have the same dimensions which depend upon D.
- An additional reduction can be obtained by fixing a *reference signal*, indexed by $r\in\mathbb{N}_D$. This is achieved replacing $\hat{\mathcal{C}}^X(0,0,0)$ with $\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}}$ defined entry-wise by:
$$(\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}})_{d_1d_2d_3}:=\left(\hat{\mathcal{C}}^X(0,0,0)\right)_{d_1d_2d_3r} . \tag{17}$$
Notice that $\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}}$ is a third order tensor whereas $\hat{\mathcal{C}}^X(0,0,0)$ is a fourth order tensor.
- $\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}}$ is a *symmetric* tensor[10]. This implies that $(\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}})_{\langle m_1\rangle}=(\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}})_{\langle m_2\rangle}$ for any $m_1,m_2\in\mathbb{N}_3$. Therefore the expression for the tensorial kernel (16) when $\hat{\mathcal{C}}_{\mathscr{L}}^X$ is replaced by $\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}}$ can be simplified to:
$$k_{\mathscr{L}}(\boldsymbol{X}_1,\boldsymbol{X}_2)=\exp\left(-\tfrac{1}{2\sigma^2}g(\boldsymbol{V}_{\mathcal{C}_{4\boldsymbol{X}_1}^{\mathrm{r}},1}^{(1)},\boldsymbol{V}_{\mathcal{C}_{4\boldsymbol{X}_2}^{\mathrm{r}},1}^{(1)})\right) , \tag{18}$$
that is, only the first (or any other) factor needs to be computed.
- Let $\boldsymbol{Z}=\boldsymbol{V}_{\mathcal{C}_{4\boldsymbol{X}_1}^{\mathrm{r}},1}^{(1)\top}\boldsymbol{V}_{\mathcal{C}_{4\boldsymbol{X}_2}^{\mathrm{r}},1}^{(1)}$. Simple matrix algebra shows that (18) is equivalent to
$$k_{\mathscr{L}}(\boldsymbol{X}_1,\boldsymbol{X}_2)=\exp\left(-\frac{1}{\sigma^2}(I_D-\mathrm{trace}(\boldsymbol{Z}^\top\boldsymbol{Z})\right) , \tag{19}$$
which is more efficiently computed.

The procedure in Algorithm 1 summarizes the steps required to evaluate the tensorial kernel for the input patterns in $\mathcal{D}_N$ when $\mathcal{L}$ is the zero-lags tuple and a reference signal is used. Similar steps are required for out-of-sample evaluations. The order reduction obtained from using a reference signal directly impacts on the computation at line (i) and at line (ii) of Algorithm 1. Note that alternative kernel methods differ in the way they use the kernel matrix $\boldsymbol{K}_r$ to carry on the estimation of the parameters $\{\alpha_n : n\in\mathbb{N}_N\}$ and b in (5). In experiments we use Least Squares Support Vector Machine (LS-SVM) classifiers, which we briefly recall in Appendix C.

D. Connection with the System Dynamics

The properties of cumulants recalled in the previous Section enable us to bridge the gap between the kernel constructed

[8] Note that in Section II-A we have required for simplicity that time-frames of multichannel signals refer to the same time instants (see equation (1)). A more general problem formulation amounts at relaxing this condition.

[9] A similar restriction is generally considered in the context of blind source separation [17].

[10] This simply means that for all $[d_1,\ d_2,\ d_3]\in\mathbb{N}_D^3$, $(\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}})_{d_1d_2d_3}=(\mathcal{C}_{4\boldsymbol{X}}^{\mathrm{r}})_{d_{P(1)}d_{P(2)}d_{P(3)}}$ where P is any permutation of the set $\{1,2,3\}$.

Algorithm 1: KERNELMATRIX($\mathcal{D}_N, \sigma, r$)

for each $n \in \mathbb{N}_N$

comment: Precompute cumulant tensors and SVD's

$$\textbf{do} \begin{cases} \mathcal{A}^{(n)} & \leftarrow \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}^{(n)}}(\text{see (17), (8), (29)}) \ (i) \\ \boldsymbol{V}_{\mathcal{A}^{(n)},1} & \leftarrow \mathrm{SVD}(\mathcal{A}^{(n)}_{<1>}) \qquad (ii) \end{cases}$$

for each $n, m \in \mathbb{N}_N$ and $m > n$

comment: Compute entries of the kernel matrix $\boldsymbol{K}$

$$\textbf{do} \begin{cases} \boldsymbol{Z} & \leftarrow \boldsymbol{V}^{\top}_{\mathcal{A}^{(n)},1}\boldsymbol{V}_{\mathcal{A}^{(m)},1} \\ (\boldsymbol{K}_r)_{nm} & \leftarrow \exp\left(-\frac{1}{\sigma^2}(I_D - \mathrm{trace}(\boldsymbol{Z}^{\top}\boldsymbol{Z})\right) \end{cases}$$

$\boldsymbol{K}_r \leftarrow \boldsymbol{K}_r + \boldsymbol{K}_r^{\top} + \boldsymbol{I}_N$

return $(\boldsymbol{K}_r)$

on the observed outputs and the dynamics of the generating systems. In particular, we have the following *invariance properties*.

Theorem IV.1 (Asymptotical invariances)**.** *Assume systems $\Sigma_1 = (\boldsymbol{A}_1, \boldsymbol{B}_1, \boldsymbol{C}_1)$ and $\Sigma_2 = (\boldsymbol{A}_2, \boldsymbol{B}_2, \boldsymbol{C}_2)$ driven by, respectively, mutually independent white ergodic processes (W_1, V_1) and mutually independent white ergodic processes (W_2, V_2). Additionally assume that V_1 and V_2 are Gaussian. Let $\boldsymbol{X}_1 \in \mathbb{R}^D \otimes \mathbb{R}^T$ and $\boldsymbol{X}_2 \in \mathbb{R}^D \otimes \mathbb{R}^T$ be given realizations from time-frames of the output processes taken from the steady-state behavior of the two systems. Then for $T \to \infty$ and any $i \in \mathbb{N}_3$*

1) *if $\boldsymbol{C}_1 = \boldsymbol{C}_2$ the evaluation of the $p-$th factor can be restated as:*

$$k^m(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_1}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_2}) = \exp\left(-\frac{1}{2\sigma^2} g\left(\boldsymbol{V}^{(m)}_{\mathcal{C}^{Z_1}(0,0,0),1}, \boldsymbol{V}^{(m)}_{\mathcal{C}^{Z_2}(0,0,0),1}\right)\right) \quad (20)$$

   and hence depends only on the fourth order cross-cumulants of the state processes.
2) *if $\boldsymbol{X}'_1, \boldsymbol{X}'_2 \in \mathbb{R}^D \otimes \mathbb{R}^T$ are also output realizations obtained from systems Σ_1 and Σ_2 respectively, then we have:*
   a) $k^m(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}'_1}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}'_2}) = k^m(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_1}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_2})$
   b) $k^m(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}'_1}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_1}) = 1$ *and* $k^m(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}'_2}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_2}) = 1$.

Proof: We indicate here by $\boldsymbol{X}$ (resp. $\boldsymbol{X}'$) the generic output process, either $\boldsymbol{X}_1$ or $\boldsymbol{X}_2$ (resp. $\boldsymbol{X}'_1$ or $\boldsymbol{X}'_2$). First of all notice that, since the output processes are ergodic, for $T \to \infty$ we have:

$$(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}})_{d_1 d_2 d_3} \to \left(\mathcal{C}^X(0,0,0)\right)_{d_1 d_2 d_3 r}$$

for any $[d_1,\ d_2,\ d_3] \in \mathbb{N}^3_D$. In particular, this implies that $\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}} \equiv \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}'}$ from which the equations in 2.a and 2.b immediately follow. In order to prove the first part notice now that, without loss of generality[11], we can consider $\boldsymbol{C}$ to be orthogonal. By Proposition III.1 we have:

$$\mathcal{C}^X(0,0,0) = \mathcal{C}^Z(0,0,0) \times_1 \boldsymbol{C} \times_2 \boldsymbol{C} \times_3 \cdots \times_{J-1} \boldsymbol{C} + \mathcal{C}^V(0,0,0)$$

where $\mathcal{C}^V(0,0,0)$ vanishes by point 2 in Proposition III.1. Denote by $\boldsymbol{A} \otimes \boldsymbol{B}$ the Kronecker product between any two matrices $\boldsymbol{A}$ and $\boldsymbol{B}$. The $i-$th mode unfolding $(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}})_{<i>}$ corresponds now to

$$\left(\mathcal{C}^X(0,0,0)\right)_{<i>} = \boldsymbol{C}\left(\mathcal{C}^Z(0,0,0)\right)_{<i>}(\boldsymbol{C} \otimes \boldsymbol{C} \otimes \boldsymbol{C})^{\top}$$

where we used a basic property of the matricization operation [13], [34]. Notice that $\boldsymbol{C} \otimes \boldsymbol{C} \otimes \boldsymbol{C}$ is an orthogonal matrix and hence if $\boldsymbol{U}^{(i)}_{\mathcal{C}^Z(0,0,0),1}\boldsymbol{S}^{(i)}_{\mathcal{C}^Z(0,0,0),1}\boldsymbol{V}^{(i)\top}_{\mathcal{C}^Z(0,0,0),1}$ is the thin SVD [22] factorization of $\left(\mathcal{C}^Z(0,0,0)\right)_{<i>}$ then for the right singular vectors $\boldsymbol{V}^{(i)}_{\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}},1}$ we have

$$\boldsymbol{V}^{(i)}_{\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}},1} = (\boldsymbol{C} \otimes \boldsymbol{C} \otimes \boldsymbol{C})\,\boldsymbol{V}^{(i)}_{\mathcal{C}^Z(0,0,0),1}. \quad (21)$$

Replacing (21) in the formula that gives $k^i(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_1}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_2})$ and exploiting the orthogonality of $\boldsymbol{C} \otimes \boldsymbol{C} \otimes \boldsymbol{C}$ now proves (20). ■

The former result is best understood in light of a concrete application that we consider next.

V. THE CASE OF EG SIGNALS

State-space methods have proven to be a valuable tool in the wide domain of neural data analysis [32]. In particular, a state-space framework for multichannel electroencephalography (EEG) signals was proposed in [4], [7] with the purpose of modeling cortical connectivity[12]. In this case (10)-(11) is assumed to describe the physics behind the cortical dynamics. Specifically, the state process Z represents the latent cortical signals. The output process X represents the evolution in time of the electric potentials (the multichannel EEG signals) measured at the scalp. Equation (11) expresses them as a noisy linear combination of the cortical signals.

A. Classification Problems Involving EG Signals

There are multiple classification problems within this context. For instance, the N training input patterns might refer to multichannel EG signals measured from N different subjects on certain time windows. In this case, each input (a time frame of the output process) is labelled according to the class to which the corresponding subject belongs. For experiments on a single subject, on the other hand, one might assume that the subject-specific system remains unchanged whereas what changes is the cortical activity driven by external stimuli belonging to different categories (different stochastic processes).

B. Interpretation of Theorem IV.1

For the case of EG signals Theorem IV.1 reads as follows. The first part means that, although the factor kernel

[11] In fact, as well known, there is an entire class of state-space representations that are equivalent up to similarity transformations.

[12] We argue that a similar state-space representation is plausible also to describe MEG signals.

$k^p(\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_1}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}_2})$ (and, as a consequence, the kernel function (18)) is defined on the measured EG signals, under the given assumptions it depends uniquely on (cumulants of) the cortical signals. Therefore it works as a similarity measure between the latent mental states. The second part of Theorem IV.1 is a consequence of the fact that the kernel is defined upon a statistic (in fact, the cumulant function) and is therefore invariant with respect to the sample signal that is observed (see also Remark IV.2). In particular, the equations in 2.b state that two time frames (for $T \to \infty$) are *maximally similar* if they are generated by the same systems (e.g. healthy brains), possibly driven by different realizations of the same stochastic process.

VI. Experimental Results

In this Section we compare the results obtained with alternative classification approaches. Our primary goal is to evaluate the performance of discriminative classifiers based on the cumulant-based kernel function (19) (denoted as sub$-\mathcal{C}$) against those obtained based on a number of alternative similarity measures. As (19), the first group of alternative kernels that we considered is also based on zero-lags cumulant tensors with a reference signal. These are:

$$k_{\mathrm{RBF}-\mathcal{C}}(\boldsymbol{X}, \boldsymbol{Y}) := \exp\left(-\frac{1}{2\sigma^2}\|\mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}} - \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{Y}}\|^2_F\right) \quad (22)$$

$$k_{\mathrm{lin}-\mathcal{C}}(\boldsymbol{X}, \boldsymbol{Y}) := \langle \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{X}}, \mathcal{C}^{\mathrm{r}}_{4\boldsymbol{Y}} \rangle \ . \quad (23)$$

Contrary to sub$-\mathcal{C}$, these kernels do not exploit the algebraic structure of tensors [37]. We also tested an instance of the Fisher kernel based upon linear Gaussian systems with state dimension d (denoted as Fisher$-d$; see Appendix A). This kernel requires the identification of models. We used the implementation of the EM algorithm for linear Gaussian systems by Zoubin Ghahramani[13]. Additionally, we considered kernels that simply act on the concatenation of the vectors corresponding to each channel:

$$k_{\mathrm{RBF}}(\boldsymbol{X}, \boldsymbol{Y}) := \exp\left(-\frac{1}{2\sigma^2}\|\boldsymbol{X} - \boldsymbol{Y}\|^2_F\right) \quad (24)$$

$$k_{\mathrm{lin}}(\boldsymbol{X}, \boldsymbol{Y}) := \langle \boldsymbol{X}, \boldsymbol{Y} \rangle \ . \quad (25)$$

Notice that the latter kernel functions do not exploit the dynamical structure of the data. Nevertheless they are widely used in the context of machine learning to deal with multivariate time series. As discriminative classifiers in all our experiments we used LSSVM classifiers (Appendix C). Finally we tested generative classifiers and considered the same model class as for the Fisher kernel. With these methods classification was obtained as follows. Assume class labels $\mathscr{Y} = \{s_1, s_2, \ldots, s_K\}$. To classify a sequence into one of the K classes, we trained K HMMs with state dimension d (denoted as HMM$-d$), one per class, and then computed the log-likelihood that each model gave to the test sequence; if the $k-$th model was the most likely, then we declared the class of the sequence to be s_k.

[13]Available at http://www.gatsby.ucl.ac.uk/~zoubin/software.html.

A. Synthetic Examples: 3-class problems

First we validate the proposed approach on synthetic examples where we generate time series according to class-specific models.

1) Generative Method: For $n \in \mathbb{N}_N$ we let y_n be drawn from a categorical distribution with set of symbols $\mathcal{Y} = \{1, 2, 3\}$ and success probabilities $p_1 = p_2 = p_3 = \frac{1}{3}$. Denote by $\mathcal{U}([a, b]^d)$ the uniform distribution on $[a, b]^d$ and by $\mathcal{N}(0_d, \boldsymbol{I}_d)$ the $d-$variate normal distribution. For any y_n we generated the corresponding training output processes $\bar{X}^{(n)}$ according to

$$\begin{aligned} Z^{(n)}(t+1) &= \boldsymbol{A}_{(n)} Z^{(n)}(t) + \boldsymbol{B} W^{(n)}(t), \\ \bar{x}^{(n)}_i(t) &= \left(\boldsymbol{C} Z^{(n)}(t)\right)^{\tau}_i + v^{(n)}_i(t), \qquad i \in \mathbb{N}_5 \end{aligned}$$

where for any n and t, $W^{(n)}(t) \sim \mathcal{U}([0, 10]^3)$ and $V^{(n)}(t) \sim \mathcal{N}(0_5, \boldsymbol{I}_5)$. We considered three distinct cases, namely $\tau = 1$ (*linear state-space models*), $\tau = 2$ and $\tau = 3$ (*nonlinear state-space models*). Additionally note that we have considered the case where each model, indexed by n, is constructed around one of the three class prototypes but does not coincide with it[14]. Specifically, the system matrices $\boldsymbol{A}_{(n)}, \boldsymbol{B}_{(n)} \in \mathbb{R}^3 \otimes \mathbb{R}^3$ and $\boldsymbol{C} \in \mathbb{R}^5 \otimes \mathbb{R}^3$ were defined as follows:

$$\left(\tilde{\boldsymbol{A}}_{(n)}\right)_{ij} = \begin{cases} \sin(ij), & \text{if } y_n = 1 \\ \cos(ij), & \text{if } y_n = 2 \\ \sin(ij + \pi)^3, & \text{if } y_n = 3 \end{cases},$$

$$R^{(n)} = \mathrm{diag}\left(\left[-0.5 + F^{(n)}_1, -0.2 + F^{(n)}_2, 0.6 + F^{(n)}_3\right]\right),$$

$$\left(\boldsymbol{B}_{(n)}\right)_{ij} = \sin(ij) + F^{(n)}_4, \ \left(\boldsymbol{C}\right)_{ij} = \cos(ij),$$

$$\boldsymbol{A}_{(n)} = \tilde{\boldsymbol{A}}_{(n)} R^{(n)} \tilde{\boldsymbol{A}}^{-1}_{(n)},$$

where for any n, $F^{(n)} \sim \mathcal{U}([-0.4, 0.4]^4)$. In generating the output measurements we let $t \in \mathbb{N}_{2000}$ and formed $\bar{\boldsymbol{X}}^{(n)} \in \mathbb{R}^5 \otimes \mathbb{R}^{600}$ based upon the last 600 time instants. Finally, for any n we took as input pattern $\boldsymbol{X}^{(n)}$ defined column-wise as[15] $X^{(n)}_i = \bar{X}^{(n)}_i - \frac{1}{600}\bar{\boldsymbol{X}}^{(n)} 1_{600}$ for $i \in \mathbb{N}_{600}$.

This scheme was used to draw the N training pairs as well as 200 input-output pairs used across different experiments for the purpose of comparing the different procedures (test set).

2) Test Procedure: For the generative classifiers we trained 3 (one per class) HMMs with state dimension d using the appropriate subset of the N training patterns. For the cumulant-based kernels we let $r = 1$, that is, we used the first output channel as the reference signal. All the kernels where used within discriminative classifiers that were trained using the N training samples via the LS-SVMlab toolbox (www.esat.kuleuven.be/sista/lssvmlab, [10]). We used the de-

[14]In many applications, in fact, it is reasonable to assume that measurements within the same class might still differ in the way they have been generated. Consider for instance the case of EG signals taken from different subjects. In this case one observes intra-patients variability even within the same class (e.g. healthy patients).

[15]It is worthy remarking at this point that the cumulants of Section III-B were defined for zero-mean vector processes.

fault one-versus-one multiclass encoding and 10-fold cross validation to perform model selection (namely the choice of tuning parameter γ in (30) and — where it applies — the kernel parameter σ). To estimate models necessary for the Fisher$-d$ kernel we used the whole set of N training pairs.

We considered different values of N. For each setting we performed 30 experiments each of which was based upon an independent $\mathcal{D}_N$. In Table I we report the mean (with standard deviation in parenthesis) of misclassification errors obtained by the different models on the test set.

TABLE I: Synthetic examples: misclassification errors on test data

$\tau = 1$ (linear state-space models)

kernel/method	$N = 80$	$N = 160$	$N = 320$	$N = 640$
sub$-\mathcal{C}$	0.57 (0.04)	0.54 (0.02)	0.51 (0.02)	0.49 (0.01)
lin$-\mathcal{C}$	0.65 (0.01)	0.64 (0.02)	0.62 (0.02)	0.61 (0.02)
RBF$-\mathcal{C}$	0.57 (0.02)	0.56 (0.02)	0.54 (0.02)	0.54 (0.01)
lin	0.67 (0.02)	0.68 (0.02)	0.68 (0.02)	0.67 (0.02)
RBF	0.64 (0.02)	0.64 (0.03)	0.61 (0.02)	0.61 (0.01)
fisher-2	0.42 (0.04)	0.59 (0.12)	0.65 (0.03)	0.66 (0.02)
fisher-3	0.38 (0.04)	0.60 (0.09)	0.65 (0.03)	0.66 (0.03)
fisher-4	0.35 (0.04)	0.56 (0.12)	0.62 (0.03)	0.66 (0.05)
HMM-2	0.38 (0.04)	0.37 (0.05)	0.38 (0.05)	0.36 (0.01)
HMM-3	**0.27** (0.03)	0.28 (0.04)	0.25 (0.02)	**0.24** (0.01)
HMM-4	**0.27** (0.03)	**0.26** (0.02)	**0.24** (0.02)	**0.24** (0.01)

$\tau = 2$ (nonlinear state-space models)

kernel/method	$N = 80$	$N = 160$	$N = 320$	$N = 640$
sub$-\mathcal{C}$	0.14 (0.03)	0.10 (0.02)	**0.07** (0.02)	**0.05** (0.01)
lin$-\mathcal{C}$	0.40 (0.12)	0.38 (0.18)	0.45 (0.18)	0.50 (0.14)
RBF$-\mathcal{C}$	0.66 (0.02)	0.67 (0.02)	0.67 (0.02)	0.67 (0.01)
lin	0.66 (0.02)	0.64 (0.03)	0.62 (0.03)	0.61 (0.03)
RBF	0.66 (0.02)	0.67 (0.02)	0.67 (0.02)	0.68 (0.01)
fisher-2	0.41 (0.24)	0.41 (0.21)	0.49 (0.05)	0.50 (0.20)
fisher-3	0.41 (0.20)	0.33 (0.17)	0.42 (0.15)	0.47 (0.14)
fisher-4	0.46 (0.13)	0.48 (0.11)	0.58 (0.15)	0.53 (0.14)
HMM-4	0.15 (0.06)	0.13 (0.05)	0.13 (0.03)	0.13 (0.01)
HMM-8	**0.10** (0.06)	**0.09** (0.04)	0.08 (0.02)	0.08 (0.01)
HMM-12	**0.10** (0.06)	**0.09** (0.04)	0.08 (0.02)	0.08 (0.01)

$\tau = 3$ (nonlinear state-space models)

kernel/method	$N = 80$	$N = 160$	$N = 320$	$N = 640$
sub$-\mathcal{C}$	**0.19** (0.03)	**0.14** (0.02)	**0.12** (0.02)	**0.10** (0.02)
lin$-\mathcal{C}$	0.54 (0.09)	0.60 (0.07)	0.63 (0.06)	0.67 (0.05)
RBF$-\mathcal{C}$	0.66 (0.02)	0.66 (0.01)	0.66 (0.02)	0.66 (0.01)
lin	0.63 (0.02)	0.62 (0.02)	0.62 (0.03)	0.61 (0.02)
RBF	0.66 (0.02)	0.66 (0.02)	0.66 (0.02)	0.66 (0.01)
fisher-2	0.42 (0.13)	0.51 (0.12)	0.58 (0.11)	0.54 (0.11)
fisher-4	0.46 (0.11)	0.48 (0.12)	0.59 (0.12)	0.59 (0.11)
fisher-8	0.32 (0.08)	0.51 (0.11)	0.50 (0.10)	0.51 (0.11)
HMM-4	0.21 (0.04)	0.21 (0.03)	0.20 (0.07)	0.20 (0.01)
HMM-8	0.21 (0.04)	0.20 (0.03)	0.19 (0.02)	0.20 (0.01)
HMM-12	0.21 (0.04)	0.20 (0.03)	0.19 (0.02)	0.20 (0.01)

Remark VI.1. *According to Table I the cumulant kernel (sub$-\mathcal{C}$) outperforms all the alternative similarity measures (including those based on generative models). Notably HMMs work better for the case where the underlying models are linear ($\tau = 1$) and otherwise perform comparably or worse ($\tau = 2$ and $\tau = 3$).*

B. Biomag 2010 Data

As a real life application we considered a brain decoding problem. The experiment was designed to investigate whether the brain activity of a subject modulating his/her attention towards a specific spatial direction (left or right), could be used as the control signal in a brain-computer interface (BCI) system [45], [27].

1) Experimental Setup and Preprocessing: Brain data were recorded by means of a MEG system and distributed by the Biomag 2010 data analysis competition [16]. Samples of 274 sensors were collected at 1200 Hz for 255 trials per subject and refer to 4 subjects [17]. Two standard pre-processing steps were operated on all time series, specifically: first downsampling from 1200 to 300 Hz and secondly band-pass filtering in order to remove frequency components outside the range $5 - 70$ Hz. The brain activity of interest related to attention tasks is known to fall within that range [45]. In this work we present results for subject 1 in the original study. Based on previous findings [45], [27], we restricted our analysis to a subset of 45 out of the 274 sensors/timeseries of the initial dataset corresponding to the parietal areas of the brain. As in [45], we considered the last 2000 ms of each trial to remove artifacts occurring immediately after the stimulus offset. The final dataset consisted of 255 trials, 127 corresponding to the *left* stimulus and 128 to the *right* stimulus. The input pattern corresponding to each trial consisted of a 45×600 matrix. Additionally, we constructed a second dataset in order to compare results against the state-of-the-art analysis found in [45]. In this case, each of the 45 channels of each trial was represented by a vector of 129 elements corresponding to values of the power spectral density (PSD) in the $5 - 70$ Hz range.

2) Test Procedure: Out of the 255 recordings available 100 were selected at random for testing purposes. Within the remaining patterns N were extracted at random in repeated experiments to form the training set $\mathcal{D}_N$. We tested the same kernel functions (22)-(25) as for the synthetic experiment above as well as the proposed kernel function (19) (sub-$\mathcal{C}$) within the LS-SVMlab toolbox. Additionally we tested the kernel function (25) (PSD-lin) on the PSD dataset described above and the Fisher kernel (Appendix A). The Areas under the ROC curves (AUCs) obtained via the best performing kernels are reported in Table II along with those obtained by generative classifiers. We reported results for Fisher$-d$ and HMM$-d$ only for $d = 2$ since $d > 2$ led to worse performances.

TABLE II: Real-life problem: AUCs measured on test data

kernel/method	$N = 90$	$N = 110$	$N = 130$	$N = 150$
sub$-\mathcal{C}$	$\mathbf{0.86}(0.01)$	$\mathbf{0.87}(0.01)$	$\mathbf{0.87}(0.01)$	$\mathbf{0.88}(0.01)$
PSD-lin	0.80(0.02)	0.81(0.01)	0.82(0.01)	0.82(0.01)
fisher-2	0.68(0.05)	0.70(0.05)	0.72(0.04)	0.73(0.05)
HMM-2	0.81 (0.02)	0.81 (0.02)	0.81 (0.02)	0.83 (0.01)

[16] The dataset was downloaded from ftp://ftp.fcdonders.nl/pub/courses/biomag2010/competition1-dataset.

[17] The Biomag dataset corresponds to subjects 1, 2, 3 and 7 respectively in [45].

VII. Conclusion

We have proposed a novel kernel function that measures similarity between multichannel signals, based upon the spectral information of cumulant tensors. These tensors are constructed directly from observed measurements and do not require specific modeling assumptions, nor a foregoing identification step. This allows one to derive fully data-driven pattern recognition methods while still exploiting the dynamical nature of the data. We have discussed computational aspects and shown properties of the proposed cumulant kernels. The significance of these properties was discussed in connection with a specific application, namely EG data analysis. Experimental results compare favorably to those obtained by discriminative classifiers built upon alternative kernel functions or by generative methods. Alternative kernels include similarities that do not exploit the dynamical structure of the data. Additionally we have considered an instance of the Fisher kernel that, similarly to an approach based on HMMs, requires the estimation of linear Gaussian models.

Acknowledgment

We thank the anonymous reviewers for helpful comments.

Research supported by Research Council KUL: CIF1 and STRT1/08/023 (2), GOA/11/05 Ambiorics, GOA/10/09 MaNet, CoE EF/05/006 Optimization in Engineering (OPTEC) en PFV/10/002 (OPTEC), IOF-SCORES4CHEM, several PhD/postdoc and fellow grants; Flemish Government: FWO: PhD/postdoc grants, projects: G0226.06 (cooperative systems and optimization), G.0427.10N, G.0302.07 (SVM/Kernel), G.0557.08 (Glycemia2), G.0588.09 (Brain-machine) research communities (WOG: ICCoS, ANMMM, MLDM). IWT: PhD Grants, Eureka-Flite+, SBO LeCoPro, SBO Climaqs, SBO POM, O&O-Dsquare. Belgian Federal Science Policy Office: IUAP P6/04 (DYSCO, Dynamical systems, control and optimization, 2007-2011). IBBT. EU: ERNSI; FP7-HD-MPC (INFSO-ICT-223854), COST intelliCIS, FP7-EMBOCON (ICT-248940). Contract Research: AMINAL.

References

[1] S.I. Amari. Natural gradient works efficiently in learning. *Neural computation*, 10(2):251–276, 1998.

[2] A. Bissacco, A. Chiuso, and S. Soatto. Classification and recognition of dynamical models: The role of phase, independent components, kernels and optimal transport. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 29(11):1958–1972, 2007.

[3] C. Bourin and P. Bondon. Efficiency of high-order moment estimates. *IEEE Trans. on Signal Processing*, 46(1):255–258, 1998.

[4] S. L. Bressler, C. G. Richter, Y. Chen, and M. Ding. Cortical functional network organization from autoregressive modeling of local field potential oscillations. *Statistics in Medicine*, 26(21):3875–3885, 2007.

[5] D. R. Brillinger and M. Rosenblatt. *Computation and interpretation of $k-th$ order spectra*, pages 189–232. Advanced Seminar on Spectral Analysis of Time Series, University of Wisconsin–Madison, 1966. John Wiley, New York, 1967.

[6] O. Cappé, E. Moulines, and T. Ryden. *Inference in Hidden Markov Models*. Springer Verlag, 2005.

[7] B. L. P. Cheung, B. A. Riedner, G. Tononi, and B.; Van Veen. Estimation of cortical connectivity from EEG using state-space models. *IEEE Trans. on Biomedical Engineering*, 57(9):2122–2134, 2010.

[8] P. Comon and C. Jutten. *Handbook of Blind Source Separation: Independent component analysis and applications*. Academic Press, 2010.

[9] C. Cortes and V. Vapnik. Support vector networks. *Machine Learning*, 20:273–297, 1995.

[10] K. De Brabanter, P. Karsmakers, F. Ojeda, C. Alzate, J. De Brabanter, K. Pelckmans, B. De Moor, J. Vandewalle, and J. A. K. Suykens. LS-SVMlab toolbox user's guide version 1.8. *Internal Report 10-146, ESAT-SISTA, K.U.Leuven (Leuven, Belgium)*, 2010.

[11] K. De Cock and B. De Moor. Subspace angles between ARMA models. *Systems & Control Letters*, 46(4):265–270, 2002.

[12] L. De Lathauwer. Characterizing higher-order tensors by means of subspaces. *Internal Report 11-32, ESAT-SISTA, K.U. Leuven (Leuven, Belgium)*, 2011.

[13] L. De Lathauwer, B. De Moor, and J. Vandewalle. A multilinear singular value decomposition. *SIAM Journal on Matrix Analysis and Applications*, 21(4):1253–1278, 2000.

[14] L. De Lathauwer, B. De Moor, and J. Vandewalle. An introduction to independent component analysis. *Journal of Chemometrics*, 14(3):123–149, 2000.

[15] A. Edelman, T. A. Arias, and S. T. Smith. The geometry of algorithms with orthogonality constraints. *SIAM Journal on Matrix Analysis and Applications*, 20(2):303–353, 1999.

[16] G. Favier and A.Y. Kibangou. Tensor-based methods for system identification. *International Journal of Sciences and Techniques of Authomatic Control Computer Engineering (IJ-STA)*, 3(1):870–889, 2009.

[17] A. Ferréol and P. Chevalier. On the behavior of current second and higher order blind source separation methods for cyclostationary sources. *IEEE Trans. on Signal Processing*, 48(6):1712–1725, 2002.

[18] J. A. R. Fonollosa. Sample cumulants of stationary processes: asymptotic results. *IEEE Trans. on Signal Processing*, 43(4):967–977, 2002.

[19] T. Gärtner. A survey of kernels for structured data. *ACM SIGKDD Explorations Newsletter*, 5(1):49–58, 2003.

[20] Z. Ghahramani and G.E. Hinton. Parameter estimation for linear dynamical systems. *University of Toronto, Technical Report CRG-TR-96-2*, page 6 pages, 1996.

[21] G. B. Giannakis, Y. Inouye, and J. M. Mendel. Cumulant based identification of multichannel moving-average models. *IEEE Trans. on Automatic Control*, 34(7):783–787, 2002.

[22] G. H. Golub and C. F. Van Loan. *Matrix Computations*. Johns Hopkins University Press, third edition, 1996.

[23] E. J. Hannan. *Multiple time series*. John Wiley & Sons, Inc. New York, NY, USA, 1971.

[24] T.S. Jaakkola and D. Haussler. Exploiting generative models in discriminative classifiers. *Advances in neural information processing systems*, pages 487–493, 1999.

[25] M. W. Kadous. *Temporal classification: extending the classification paradigm to multivariate time series*. PhD thesis, School of Computer Science and Engineering, The University of New South Wales, 2002.

[26] T. Kailath, A. H. Sayed, and B. Hassibi. *Linear estimation*. Prentice Hall Upper Saddle River, NJ, USA, 2000.

[27] S. P. Kelly, E. Lalor, R. B. Reilly, and J. J. Foxe. Independent brain computer interface control using visual spatial attention-dependent modulations of parieto-occipital alpha. In *Conference Proceedings of $2-nd$ International IEEE EMBS Conference on Neural Engineering*, pages 667–670, 2005.

[28] J. Liang and Z. Ding. Blind MIMO system identification based on cumulant subspace decomposition. *IEEE Trans. on Signal Processing*, 51(6):1457–1468, 2003.

[29] R. J. Martin. A metric for ARMA processes. *IEEE Trans. on Signal Processing*, 48(4):1164–1170, 2002.

[30] C. L. Nikias and J. M. Mendel. Signal processing with higher-order spectra. *Signal Processing Magazine, IEEE*, 10(3):10–37, 2002.

[31] C. L. Nikias and A. P. Petropulu. *Higher-order spectra analysis: a nonlinear signal processing framework*. Prentice Hall Upper Saddle River, NJ, USA, 1993.

[32] L. Paninski, Y. Ahmadian, D.G. Ferreira, S. Koyama, K. Rahnama Rad, M. Vidne, J. Vogelstein, and W. Wu. A new look at state-space models for neural data. *Journal of computational neuroscience*, 29(1):107–126, 2010.

[33] M. B. Priestley. *Spectral analysis and time series*. Academic Press, 1981.

[34] P. A. Regalia and S. K. Mitra. Kronecker products, unitary matrices and signal processing applications. *SIAM review*, 31(4):586–613, 1989.

[35] S. Roweis and Z. Ghahramani. A unifying review of linear gaussian models. *Neural computation*, 11(2):305–345, 1999.

[36] J. Shawe-Taylor and N. Cristianini. *Kernel Methods for Pattern Analysis*. Cambridge University Press, 2004.

[37] M. Signoretto, L. De Lathauwer, and J. A. K. Suykens. A kernel-based framework to tensorial data analysis. *Neural Networks*, 24(8):861–874, 2011.

[38] I. Steinwart and A. Christmann. *Support vector machines*. Springer Verlag, 2008.

[39] J. A. K. Suykens, C. Alzate, and K. Pelckmans. Primal and dual model representations in kernel-based learning. *Statistics Surveys*, 4:148–183 (electronic). DOI: 10.1214/09–SS052, 2010.
[40] J. A. K. Suykens and J. Vandewalle. Least squares support vector machine classifiers. *Neural Processing Letters*, 9(3):293–300, 1999.
[41] J.A.K. Suykens, T. Van Gestel, J. De Brabanter, B. De Moor, and J. Vandewalle. *Least squares support vector machines*. World Scientific, 2002.
[42] A. Swami, G. Giannakis, and S. Shamsunder. Multichannel ARMA processes. *IEEE Trans. on Signal Processing*, 42(4):898–913, 2002.
[43] A. Swami and J. M. Mendel. Time and lag recursive computation of cumulants from a state-space model. *IEEE Trans. on Automatic Control*, 35(1):4–17, 1990.
[44] K. Tsuda, S. Akaho, M. Kawanabe, and K.R. Müller. Asymptotic properties of the fisher kernel. *Neural Computation*, 16(1):115–137, 2004.
[45] M. Van Gerven and O. Jensen. Attention modulations of posterior alpha as a control signal for two-dimensional brain-computer interfaces. *Journal of Neuroscience Methods*, 179(1):78–84, April 2009.
[46] S. V. N. Vishwanathan, A. J. Smola, and R. Vidal. Binet-Cauchy kernels on dynamical systems and its application to the analysis of dynamic scenes. *International Journal of Computer Vision*, 73(1):95–119, 2007.
[47] H. O. A. Wold. *A study in the analysis of stationary time series*. Almqvist & Wiksell, 1954.

APPENDIX

A. Fisher Kernel for Linear Gaussian Dynamical Systems

Suppose a collection $\mathcal{D}_N$ of labelled time frames of vector processes is given (see (3)). In order to explicitly model $p(\boldsymbol{X})$ a viable way is to assume that $\boldsymbol{X} = [X(T_0),\ X(1+T_0),\ \cdots,\ X(T-1+T_0)]$ is extracted from the output process X of a linear time-invariant dynamical system in state-space form:

$$\begin{array}{rcl} Z(t+1) & = & \boldsymbol{A}Z(t) + W(t) \\ X(t) & = & \boldsymbol{C}Z(t) + V(t)\ . \end{array} \tag{26}$$

For any $t \in \mathbb{N}$, it is assumed here that $W(t)$ and $V(t)$ are zero-mean Gaussian random vectors with covariance matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$ respectively; the initial state $X(1)$ is also assumed to be a Gaussian random vector with mean and covariance denoted respectively by M_1 and $\boldsymbol{V}_1$. The estimation of the model's parameter $\theta = \left\{\boldsymbol{A}, \boldsymbol{C}, \boldsymbol{Q}^{-1}, \boldsymbol{R}^{-1}, M_1, \boldsymbol{V}_1\right\}$ starting from multiple time frames of observed outputs $\left\{\boldsymbol{X}^{(1)}, \boldsymbol{X}^{(2)}, \ldots, \boldsymbol{X}^{(n)}\right\}$ can be performed by maximizing the log-likelihood

$$\mathcal{L}(\theta) = \log P(\{\boldsymbol{X}^{(1)}, \boldsymbol{X}^{(2)}, \ldots, \boldsymbol{X}^{(n)}\}|\theta) = \sum_{n \in \mathbb{N}_N} \log P(\boldsymbol{X}^{(n)}|\theta)\ .$$

In practice, under the aforementioned assumptions $\hat{\theta} = \arg\max \mathcal{L}(\theta)$ can be found using EM algorithms, see [20], [35]. The rationale behind the Fisher kernel now specializes as follows. Two arbitrary time frames $\boldsymbol{X}^{(i)}$ and $\boldsymbol{X}^{(j)}$ should be similar if the corresponding *Fisher scores* (which are *natural gradients* [1]) $\nabla_\theta \log P(\boldsymbol{X}^{(i)}|\theta)$ and $\nabla_\theta \log P(\boldsymbol{X}^{(j)}|\theta)$ are similar for $\theta = \hat{\theta}$. In particular, the *practical Fisher kernel* [36] is defined as

$$k_F(\boldsymbol{X}^{(i)}, \boldsymbol{X}^{(j)}) = \left(\nabla_\theta \log P(\boldsymbol{X}^{(i)}|\theta)^\top \nabla_\theta \log P(\boldsymbol{X}^{(j)}|\theta)\right)\Big|_{\theta=\hat{\theta}}\ . \tag{27}$$

In Section VI, we call the latter the Fisher$-d$ kernel to emphasize that d is the dimensionality of X in the state-space model (26). Note that $\nabla_\theta \log P(\boldsymbol{X}^{(i)}|\theta)$ can be obtained from the gradient of the expected log-likelihood involved in the M-step [6, Proposition 10.1.4]. A closed form expression for the latter is found in [20].

B. Formulas of third and fourth order cumulants

Assume a real $D-$variate discrete-time stationary vector process $\{X(t)\}$ and suppose $E[X(t)] = 0$. Then the third order cross-cumulant between the d_1-th, d_2-th and d_3-th entries of the vector process X is:

$$c^X_{d_1 d_2 d_3}(l_1, l_2) := E\left[x_{d_1}(t) x_{d_2}(t+l_1) x_{d_3}(t+l_2)\right]\ . \tag{28}$$

The fourth order cross-cumulant between the d_1-th, d_2-th, d_3-th and d_4-th entries of the vector process X is:

$$\begin{array}{l} c^X_{d_1 d_2 d_3 d_4}(l_1, l_2, l_3) := \\ \quad E\left[x_{d_1}(t) x_{d_2}(t+l_1) x_{d_3}(t+l_2) x_{d_4}(t+l_3)\right] \\ \quad - E\left[x_{d_1}(t) x_{d_2}(t+l_1)\right] E\left[x_{d_3}(t+l_2) x_{d_4}(t+l_3)\right] \\ \quad - E\left[x_{d_1}(t) x_{d_3}(t+l_2)\right] E\left[x_{d_4}(t+l_3) x_{d_2}(t+l_1)\right] \\ - E\left[x_{d_1}(t) x_{d_4}(t+l_3)\right] E\left[x_{d_2}(t+l_1) x_{d_3}(t+l_2)\right]\ . \end{array} \tag{29}$$

C. Model Estimation via LS-SVM

Within most of discriminative methods finding a classifier involves estimating the parameters $\{\alpha_n\ :\ n \in \mathbb{N}_N\}$ and b in (5). In the context of Support Vector Machines (SVM's) and related estimators [41], [38], [39], (5) is the *dual representation* that is obtained via a Lagrangian duality argument from a *primal model*. Vapnik's original SVM formulation [9] translates into convex quadratic programs. In contrast, in the Least-Squares SVM (LS-SVM) classifier [41], a modification of the SVM primal problem leads to a considerably simpler estimation problem. The latter corresponds to find the solution (α, b) of the following system of linear equations [40]:

$$\left[\begin{array}{cc} 0 & Y^\top \\ Y & \boldsymbol{\Omega} + \frac{1}{\gamma}\boldsymbol{I}_N \end{array}\right] \left[\begin{array}{c} b \\ \alpha \end{array}\right] = \left[\begin{array}{c} 0 \\ 1_N \end{array}\right] \tag{30}$$

where γ is a user-defined parameter, $\mathbb{R}^N \ni 1_N = (1, 1, \ldots, 1)$, $\boldsymbol{\Omega} \in \mathbb{R}^N \otimes \mathbb{R}^N$ is defined entry-wise by $(\boldsymbol{\Omega})_{ij} = y_i y_j (\boldsymbol{K}_r)_{ij}$ and $\boldsymbol{K}_r$ is a kernel matrix such as the one computed in Algorithm 1.