 doi:10.20944/preprints201912.0340.v1

Article

EEG signal classification: an application to the emotion-related brain anticipatory activity.

Marco Bilucaglia [1†], Gian Marco Duma [2†], Giovanni Mento[2], Luca Semenzato [2] and Patrizio Tressoldi [2*]

[1] Behavior and BrainLab, Università IULM, marco.bilucaglia@iulm.it
[2] Department of Department of Developmental Psychology, University of Padova, Padova, Italy, gianmarco.duma@phd.unipd.it
[2] Department of General Psychology, Padova University, Giovanni.mento@unipd.it
[2] Department of General Psychology, Padova University, luca.semenzato@unipd.it
[2] Department of General Psychology, Padova University, patrizio.tressoldi@unipd.it
* Correspondence: Patrizio.tressoldi@unipd.it; Tel.: (+39-049-8276623)
† Author contribution: Formal analysis, Marco Bilucaglia and Gian Marco Duma; Funding acquisition, Patrizio Tressoldi; Methodology, Giovanni Mento; Software, Luca Semenzato

Abstract: Machine Learning (ML) approaches have been fruitfully applied to several classification problems of neurophysiological activity. Considering the relevance of emotion in human cognition and behaviour, ML found an important application field in emotion identification based on neurophysiological activity. Nonetheless, the literature results present a high variability depending on the neuronal activity measurement, the signal features and the classifier type. The present work aims to provide new methodological insight on ML applied to emotion identification based on electrophysiological brain activity. For this reason, we recorded EEG activity while emotional stimuli, high and low arousal (auditory and visual) were provided to a group of healthy participants. Our target signal to classify was the pre-stimulus onset brain activity. Classification performance of three different classifiers (LDA, SVM and kNN) was compared using both spectral and temporal features. Furthermore, we also contrasted the classifiers performance with static and dynamic (time evolving) features. The results show a clear increased in classification accuracy with temporal dynamic features. In particular, the SVM classifiers with temporal features showed the best accuracy (63.8 %) in classifying high vs. low arousal auditory stimuli.

Keywords: emotion recognition; EEG signal decoding; brain anticipatory activity; machine learning; emotion related brain activity.

1. Introduction

In last decades, the vision of the brain has moved from a passive stimuli elaborator to an active reality builder. In other words, the brain is able to extract the information from the environment, building up inner models of external reality. These models are used to optimize the behavioural outcome to react to upcoming stimuli [1–4].

One of the main theoretical model assumes that the brain, in order to regulate body reaction, runs an internal model of the body in the world, as described by embodied simulation framework [5]. 'An increasingly popular hypothesis is that the brain's simulations function as Bayesian filter for incoming sensory input, driving action and constructing perception and other psychological phenomena, including emotion' [6]. In light of this, it is possible to consider emotions, not only as a reaction to the external world, but also as partially shaped by our internal representation of the environment, which help us to anticipate possible scenarios and therefore to regulate our behaviour.

The construction model of emotion [7] argues that the human being actively builds-up his/her emotions in relation to the everyday life and social context in which is placed. We actively generate a familiar range of emotions in our reality, based on their usefulness and relevance in our

 doi:10.20944/preprints201912.0340.v1

environment. In this scenario, in a familiar context we are able to anticipate which emotions will be probably elicited, depending on our model. As a consequence, the study of the anticipation of/preparation to forthcoming stimuli, may represent a precious window on the understanding of the individual internal model and emotion construction process, resulting in a better understanding of human behaviour.

A strategy to study preparatory activity could be related to experimental paradigm in which cues are provided regarding the forthcoming stimuli, allowing in this way the investigation of the brain activity dedicated to the elaboration of incoming stimuli [8,9]. Cue experimental paradigm predicting the emotional valence of the forthcoming stimuli showed that the brain anticipatory activation facilitates for example successful reappraisal via reduced anticipatory prefrontal cognitive elaboration and better integration of affective information in paralimbic and subcortical system [10]. Furthermore, preparation to forthcoming emotional stimuli has relevant implication also for psychological clinical conditions, as mood disorders or anxiety [11,12].

Recently the study of brain anticipatory activity has been extended to statistically unpredictable stimuli [13–15], providing experimental hints of specific anticipatory activity before stimuli randomly presented. Starting from the abovementioned studies, we focused on the extension of brain anticipatory activity to statistically unpredictable emotional stimuli.

According to the so-called dimensional model, the emotion can be defined in terms of 3 different attributes (or dimensions): valence, arousal and dominance. Valence measures the positiveness (ranging from unpleasant to pleasant), arousal measures the activation level (ranging from boredom to frantic excitement) and dominance measures the controllability (i.e. the sense of control) [16].

Emotions can be estimated from various physiological signals [17], such as the skin conductance, the EEG and the EEG. The latter has been received a considerable amount of attention in the last decade, introducing in the emotion recognition field several Machine Learning (ML) and signal processing techniques, originally developed in other contexts, such as the Brain-Computer Interfaces [18]. Emotion recognition has been re-drawn as a Machine Learning problem, where proper EEG-related features are used as input to specific classifiers.

The most common features belong the spectral domain, in the form of spectral powers in delta, theta alpha and gamma bands [19], as well as Power Spectral Density (PSD) bins [20]. The remaining belong to the time domain, in the form of Event-Related De/Synchronizations (ERD/ERS) and Event-Related Potentials (ERP) [19], as well as shape-related indices such as the Hjorth Parameters and the fractal dimension [20].

The most commonly used classifier is the Support Vector Machine (SVM) with Radial Basis Function (RBF) kernel, followed by the k-nearest neighbour (kNN) and the Linear Discriminant Analysis (LDA) [19,20]. Finally, most of the classifiers are implemented as not-adaptive (i.e. static) [19], in contrast to the dynamic versions that take in account the temporal variability of the features [21].

The classification performances are very variable, because of the different features and classifiers adopted. The following examples are taken from [19] - in particular, from the subset (17 over 63) of reviewed papers that focused on the arousal classification. Using an SVM (RBF kernel) and spectral features (e.g. STFT), Lin and colleagues obtained 94.4% of accuracy (i.e. percentage of corrected classification) [22], while using similar spectral features (e.g. PSD) and classifier (SVM with no kernel) Koelstra and colleagues obtained 55.7% [23]. Liu and Sourina obtained 76.5% using temporal features (e.g. fractal dimension) with a SVM (no kernel) [24], while Murugappan and Murugappan obtained 63% using similar temporal features and a SVM with polynomial kernel [25]. Finally, Thammasan and collegues obtained 85.3% using spectral features (e.g. PSD), but with a kNN (with k=3) [26]. All the classifiers were static.

The purpose of the present work is to provide new methodological advancements on the ML classification of emotions, based on the brain anticipatory activity. For this purpose, we compared the performances of tree different classifiers (namely LDA, SVM, kNN) trained using two types of EEG features (namely, spectral and temporal). In addition, each classifier was dynamically-trained, to take in account the temporal variability of the features. The results provided useful insights

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

regarding the best classifier-features configuration to better discriminate emotion-related brain anticipatory activity.

The paper is organized as follows: in Section 2, we provided a brief but comprehensive and self-contained starting point on Machine Learning; in Section 3, we described experiment design, the data processing and the classification processes; finally, in Sections 5-6 we presented, respectively, the results and the discussion.

2. The Machine Learning (ML)

In this section we briefly discuss the Machine Learning (ML) and the general classification problem, as well as some classification algorithms and feature selection methods. We are aware that the treatment is far from being fully exhaustive, but we hope it will serve as a comprehensive and self-contained starting point for novice readers. A more complete and precise treatment can be found in textbooks, such as [27,28].

2.1. The classification

The classification aims to assign a discreate labels (also known as class) to a m-dimensional numeric instance (also known as feature vector or, simply, feature) $\mathbf{x} \in \mathbb{R}^m$ by means of a function $f: \mathbb{R}^m \to Y \in \mathbb{Z}$ (also known as classifier).

Without loss of generality, will concentrate only on binary classification (class C_1 versus C_2). The multiclass classification can be performed using a cascade of binary classifications, following either the OVR (one-versus-rest) or the OVO (one-versus-one) strategies. In the OVR, each class C_k is classified against the corresponding aggregated class $\tilde{C}_k = \cup_{i \neq k} C_i$, while in the OVO each class C_k is tested against each class $C_{i \neq k}$ [28] (pp. 182-183).

The classes C_1 and C_2 are usually numerically coded as, respectively, $+1$ and -1. The classification can be, thus, reformulated into finding a discriminant function $h: \mathbb{R}^m \to \mathbb{R}$ such that:

$$f(\mathbf{x}) = \text{sign}\{h(\mathbf{x})\} \equiv \begin{cases} +1, & h(\mathbf{x}) > 0 \\ -1, & h(\mathbf{x}) < 0 \end{cases} \quad (1)$$

The discriminant function can be either linear or not-linear, corresponding to, respectively, linear or not-linear classifiers.

The so-called Bayes classifier represent the optimum (i.e. the best of all the possible classifiers), since gives the lowest conditional classification risk [27] (pp. 24-25) (i.e. the effects of a wrong classifications) and classification error (i.e. the probability of wrong classifications) [27] (pp. 26-27). It is a non-linear classifier with a discriminant function h given by [27] (p. 31):

$$h(\boldsymbol{x}) = \Pr\{C_1|\mathbf{x}\} - \Pr\{C_2|\mathbf{x}\} \quad (2)$$

where $\Pr\{C_1|\mathbf{x}\}$ and $\Pr\{C_2|\mathbf{x}\}$ are the posterior probabilities of, respectively, classes C_1 and C_2. Thanks to the Bayes theorem and applying a logarithmic transform, eq. (2) can be equivalently re-written as [27] (p. 31): (3)

$$h(\mathbf{x}) = \ln\{p_1(\mathbf{x})\} - \ln\{p_2(\mathbf{x})\} + \ln\{\wp_1\} - \ln\{\wp_2\}$$

The functions $p_1(\mathbf{x}) \doteq p(\mathbf{x}|C_1)$ and $p_2(\mathbf{x}) \doteq p(\mathbf{x}|C_2)$ are the two class-conditional probability density functions (PDF), namely the PDFs of the feature vectors belonging to classes C_1 and C_2, while the quantities $\wp_1$ and $\wp_2$ are the *a piori* probabilities for classes C_1 and C_2.

The knowledge of both the class-conditional PDFs and the a priori probabilities, or the posterior probabilities allows a direct implementation of the Bayes classifier using eq. (2-3) and it is a statistical problem. Unfortunately, in practice, the available information is rarely (if ever) such complete.

The theory of Machine Learning (ML) addresses this problem using a "learning from examples" approach: the classifier is built (or trained) using a so-called training set, a finite collection of prototypical and annotated (i.e. with the class information) features.

The so-called informative (or discriminative) classifiers estimate from the training set the class-conditional PDFs or prior probabilities [29]. More in dept, the so-called informative parametric

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

classifiers assumes a specific class-conditional PDF (e.g. multivariate gaussian PDF) whose parameters (e.g. mean and covariance matrix) are estimated from the training set. The informative non-parametric classifiers, on the contrary, do not assume any specific PDF [27] (pp. 84, 161).

Once found, the class-conditional PDFs or prior probabilities are inserted (plugged) into, respectively, eq. (2) or (3) in order to directly implement the Bayes classifier: for this reason, informative classifiers are also called plug-in classifiers [30].

Finally, the so-called generative classifiers directly implement a discriminant function by maximizing a proper criterion (e.g. the so-called Marginal Risk function, as shown the SVM section) on the whole domain of the training set [29].

2.1.1. LDA

The linear classifier, also known as Linear Discriminant Analysis (LDA) [27] (p. 117), is an example of informative parametric classifier. It corresponds to a discriminant function h that is a linear-affine combination of the feature, in the form:

$$h(\mathbf{x}) = b + \sum_{k=1}^{m} w_k x_k \equiv \mathbf{w}^T\mathbf{x} + b \tag{4}$$

where $\mathbf{w} \in \mathbb{R}^m$ and $b \in \mathbb{R}$ are the classifier's parameter, also known as projection vector and threshold, respectively.

Assuming the class-conditional PDFs as multivariate gaussian with mean values $\mathbf{m}_1$, $\mathbf{m}_2$ and equal covariance matrices $\mathbf{S}_1 = \mathbf{S}_2 \equiv \mathbf{S}$, as well as equal a priori probabilities ($\wp_1 = \wp_2$), the Bayes classifier of eq. (2) reduces to a linear classifier, with parameters [27] (p. 39):

$$\begin{aligned} \mathbf{w} &= \mathbf{S}^{-1}(\mathbf{m}_1 - \mathbf{m}_2) \\ b &= \frac{1}{2}\mathbf{w}^{\boldsymbol{T}}(\mathbf{m}_1 + \mathbf{m}_2) \end{aligned} \tag{5}$$

Both sample means $\mathbf{m}_1$, $\mathbf{m}_2$ and the sample covariance matrix $\mathbf{S}$ of eq. (4) are estimated using a maximum-likelihood approach [27] (p. 89):

$$\begin{aligned} \tilde{\mathbf{m}}_i &= \frac{1}{N_i}\sum_{\mathbf{x}\in C_i}\mathbf{x} \\ \tilde{\mathbf{S}} &= \frac{1}{N}\sum_{\mathbf{x}\in(C_1\cup C_2)}(\mathbf{x}-\tilde{\mathbf{m}}_i)(\mathbf{x}-\tilde{\mathbf{m}}_i)^T \end{aligned} \tag{6}$$

where N_i is the number of training features belonging to the class C_i and $N = N_1 + N_2$.

As shown in eq. (4), the covariance matrix $\tilde{\mathbf{S}}$ must be invertible (i.e. non-singular) but, unfortunately, this requirement is not always met. Regardless of the training set, for example, when the number of training features are less than the feature dimensionality (i.e. $N_i < m + 2$), $\tilde{\mathbf{S}}$ is certainly singular. To overcome to this problem, several LDA variations have been proposed. In the Regularized LDA (RLDA), $\tilde{\mathbf{S}}$ is replaced by its shrinkage estimation (also known as ridge estimation) defined as $\tilde{\mathbf{S}}_R(\lambda) = \lambda\tilde{\mathbf{S}} + \lambda\mathbf{I}$, where λ is the shrinkage parameter and $\mathbf{I}$ is the identity matrix. In the pseudo-inverse LDA, the inverse $\tilde{\mathbf{S}}^{-1}$ is replaced by the pseudoinverse $\tilde{\mathbf{S}}^{+}$[31].

2.1.2. SVM

The Support Vector Machine (SVM) is an example of discriminative classifier with a linear discriminant function (a linear discriminative classifier). Before going into the classifier's details, we need to explain some introductory concepts.

Geometrically, a discriminant function $\mathbf{w}^T\mathbf{x} + b$ defines a hyperplane in the m-dimensional space with equation $\Pi = \{\mathbf{x} \in \mathbb{R}^m | \mathbf{w}^T\mathbf{x} + b = 0\}$. For each training feature $\mathbf{x}_i$, a signed distance from the hyperplane Π is defined as $d_i = \mathbf{w}^T\mathbf{x}_i + b$. The minimum positive distance d_+ refers to the minimum distance of the feature vectors belonging to C_1, while the minimum negative distance d_- refers to the minimum distance of the feature vectors belonging to C_2. Training features with

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

distances equal to the minimum distance (either positive of negative) are called support vectors. Finally, the margin of the classifier is defined as the sum of the minimum positive and negative distances [27] (pp. 259-263).

In the so-called hard margin SVM, the training set is assumed linearly separable, that is, there exist a proper linear discriminant function that correctly classify each feature vector $\mathbf{x}_i$. The projection vector and the threshold are obtained by maximizing the margin, solving the following quadratic optimization problem [32]:

$$\begin{aligned} &\min_{\mathbf{w},b}\left\{\frac{\|\mathbf{w}\|^2}{2}\right\} \\ &\text{s. t.} \\ &y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 \geq 0, i = 1, \ldots, N \end{aligned} \tag{8}$$

where $y_i = \{-1, +1\}$ is the class of the i-th feature and N is the cardinality of the training set.

Margin maximization is equivalent to the minimization of the criterion R, called Expected Risk function, defined as [32]:

$$R = \frac{1}{2N}\sum_{i=1}^{N}|y_i - \mathbf{w}^T\mathbf{x}_i - b| \tag{9}$$

The expected risk function represents the mean classification error. It depends from both the training set and the chosen hyperplane and does not require any probabilistic assumption on the training set.

Hard margin SVM can be modified (the so-called SVM with soft margins) to work also with not linear-separable dataset, by adding a set of non-negative slack variables $\{\xi_i \geq 0\}_{i=1}^{N}$.
They take in account each misclassification error: if the feature $\mathbf{x}_i$ is correctly classified, the corresponding slack variable ξ_i is set to zero, otherwise its value is changed accordingly. The optimization problem for SVM with soft margins is the following [32]:

$$\begin{aligned} &\min_{\mathbf{w},b}\left\{\frac{\|\mathbf{w}\|^2}{2} + C\left(\sum_{i=1}^{N}\xi_i\right)\right\} \\ &\text{s. t.} \\ &y_i(\mathbf{w}^T\mathbf{x}_i + b) - 1 + \xi_i \geq 0, i = 1, \ldots, N \end{aligned} \tag{10}$$

where C is an additional parameter that control the penalty of the classification errors.

SVM with soft margins is not the only solution to treat not linear-separable data sets. By projecting the features into a potentially much higher dimensional space, the dataset can be made linear-separable. The projection is obtained by using a not-linear function $\mathbf{\Phi}: \mathbb{R}^m \to \mathbb{R}^{n>m}$. In estimating the classifier parameters, the optimization problems are thus solved in the (higher dimensional) transformed space and the following classification is performed on the (higher dimensional) transformed feature vectors $\mathbf{\Phi}(\mathbf{x})$.

Working with high dimensional data generally increase both the complexity of the estimation problems (the so-called curse of dimensionality) and the computational cost of the classification. If the former problem (estimation complexity) is mitigated by the fact that in the higher dimensional space the selected classifier is much simpler (linear), the latter remains [33]. In fact, as shown in eq. (4), the linear classification of each feature vector requires $O(n)$ additions and multiplications, corresponding to the scalar product $\mathbf{w}^T\mathbf{x}$.

Fortunately, for certain mapping $\mathbf{\Phi}$, the scalar product $\mathbf{\Phi}(\mathbf{x})^T\mathbf{\Phi}(\mathbf{y})$ can be equally expressed in terms of a kernel function $\kappa: \mathbb{R}^n \times \mathbb{R}^n \to \mathbb{R}$ such that [33]:

$$\mathbf{\Phi}(\mathbf{x})^T\mathbf{\Phi}(\mathbf{y}) = \kappa(\mathbf{x}, \mathbf{y}) \tag{11}$$

The property described in eq. (13) is often called kernel trick. Once defined a proper kernel κ, the scalar product can be implicitly computed in the higher dimensional space without explicitly using (or even knowing) the mapping function $\mathbf{\Phi}$. This allows a straightforward re-formulation of all the linear classification algorithms in the higher dimensional space: SVM, for example, generalize to the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

so-called Nonlinear SVM (also known as SVM with kernel). A commonly used kernel is the gaussian kernel (or Radial Basis Function, RBF), defined as [32]:

$$\kappa(\mathbf{x},\mathbf{y}) = e^{-\frac{\|\mathbf{x}-\mathbf{y}\|^2}{2\sigma^2}} \tag{12}$$

2.1.3. kNN

k-Nearest Neighbours (kNN) is an example of informative non-parametric classifiers. It estimates the posterior probabilities $\Pr\{C_1|\mathbf{x}\}$ and $\Pr\{C_2|\mathbf{x}\}$ in order to implement the Bayesian discriminant function of eq. (2).

The k-neighborhood of a feature $\mathbf{z}$, $\mathcal{B}_k(\mathbf{z})$, is defined as the first k training features closest (according to a proper distance function) to the feature $\boldsymbol{z}$. In kNN, the posterior probabilities $\Pr\{C_{1,2}|\mathbf{x}\}$ are estimated from every neighbourhood $\mathcal{B}_k(\mathbf{x}), \forall \mathbf{x} \in \mathbb{R}^m$ as the ratio of neighbours belonging to class $C_{1,2}$[34]:

$$\Pr\{C_{1,2}|\mathbf{x}\} = \frac{\#\{\mathbf{y} \in \mathcal{B}_k(\mathbf{x})|\mathbf{y}\sim C_{1,2}\}}{k} \tag{13}$$

where the notation $\mathbf{y}\sim C_k$ stands for "$\mathbf{y}$ belongs to class C_k".

As described in eq. (13), kNN training (i.e. posterior probabilities estimation) coincide also with the kNN classification: an unknown feature $\mathbf{x}$ is assigned to the class most frequently represented among all its k nearest sample features [27] (182).

The proper choice of k influences the kNN's performances. Intuitively, as k increases, the classification error gets closer to the minimum (i.e. the Bayes error), since the probability estimation becomes *globally* more accurate. On the other hand, as k increases, also the risk to include into $\mathcal{B}_k(\mathbf{x})$ very distant training features increases and the estimation becomes *locally* less accurate [27] (p. 184).

For the special case of $k = 1$, the classification error of the 1NN classifier can be bounded by the Bayes error ϵ_B as: $\epsilon_B \leq \epsilon_{1NN} \leq 2\epsilon_B$ [27] (p. 179).

As previously mentioned, kNN build the k-neighbourhood according to a proper distance or, more correctly, a proper metric. For this reason, kNN is also defined a metric classifier. A commonly used class of metrics is the Minkowsky distance d_p, defined as [27] (p. 188):

$$d_p(\mathbf{a},\mathbf{b}) = \left\{\sum_{i=1}^{m} |a_i - b_i|^p\right\}^{\frac{1}{p}} \tag{14}$$

where $p \in \mathbb{N}^+$. In particular, d_1 is called Manhattan (or city block) distance, d_2 is called Euclidean distance, while $d_\infty \doteq \max_{i=1,\dots m}\{|a_i - b_i|\}$ is called infinite-norm distance.

2.2. Classification performances

In introducing the Pattern Recognition, we underlined that the classifiers are built using a set of previously annotated class-prototypical features, the training set. It is common practice to extract from the training set a subset of annotated feature (the test set) and use it to evaluate the performances of the trained classifiers – but not to train it.

Since the training set is limited, the specific train/test splitting introduce a bias in both the training and performance evaluation. This can be avoided following the so-called k-fold cross-validation scheme. The original training set D is partitioned into k disjoint and equal sized sets, $D = \bigcup_{i=1}^{k} D_k$. The classifier is, then, trained k-times using, each time, as test set a different partition D_j and as training set the remaining $\bigcup_{i\neq j} D_i$. Finally, the overall performance are computed as the average over the k single performances [27] (pp. 483-485).

With the general term performance, we mostly refer to the classification accuracy ACC, defined as the ratio between the number of correctly classified features and the total number of features. Introducing the chance level accuracy ACC_0 as the ratio between the number of features for each class (i.e. how much balanced is the training set), we can additionally define as performance the Kappa statistic: $\kappa = (ACC - ACC_0)/ACC_0$ [35].

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

Compared to ACC, κ is a more robust performance measure, since it is normalized by the class unbalances. Another solution to take in account the class unbalances is to compare (using e.g. a t-test) the k cross-validated accuracies against k random accuracies, obtained from a random classifier [35].

2.3 Dynamic classifiers

To classify a time-varying signal (i.e. to perform a dynamic classification), an ordered sequence of features $\{\mathbf{x}_i\}_{i=1}^{N}$ (i.e. temporal features), corresponding to N adjacent temporal windows, is extracted. The temporal features are fed into either a "dynamic" classifiers, such as Hidden Markow Model (HMM) [21], or an ordered sequence of "static" classifiers $\{f_i\}_{i=1}^{N}$ [36–39]. The former fully takes in account the signal's temporal variability, since it uses the entire sequence during the training phase. The latter train each static classifier f_i, using only the corresponding features $\mathbf{x}_i$, but provides an ordered sequence of accuracies $\{ACC_i\}_{i=1}^{N}$, where each ACC_i corresponds to f_i.

2.4. Feature selection

As stated in the previous sections, the curse of dimensionality arises when the number of available training features is small, compared to the feature dimension m. In such situations, the parameter estimation becomes problematic (see e.g. the problem of the singularity of the estimated covariance matrix described in section 2.1.1) and the trained classifier usually underperforms.

As a rule of thumb, the number of training features N should be an exponential function of the dimensionality (e.g. $N = 10^m$), with a ratio growing with the complexity of the classifiers [30]: by fixing the feature dimension m, linear classifiers requires, for example, a less numerous training set. Additionally, even with an adequate training set, feature dimensionality impact on both the training and classification speed. In fact, as stated in sect. (2.1.2), linear classification requires $O(m)$ multiplications and sums to compute each scalar product. Reducing the feature dimensionality by means of so-called feature selection algorithms, a classifier can be made more robust (i.e. less sensitive to the curse of dimensionality) and efficient (in terms of computational speed).

Feature selection can be broadly described as a mapping function $s: \mathbb{R}^m \to \mathbb{R}^n$ such as:

$$s(\mathbf{x}) = \left(x_{s_1}, x_{s_2}, \dots, x_{s_n}\right)^T \tag{13}$$

where $n < m$ and $\{s_1, s_2, \dots, s_n\} \subset \{1,2, \dots, m\}$. In other words, a feature selection algorithm performs a projection of the original feature vector onto a lower dimensional subspace defined by a subset of scalar features. The best subspace, as selected among all the possible 2^m, should not significantly decrease the classification performances, both globally (i.e. how features are classified, overall) and locally (i.e. how the single feature is classified) [40].

Features selection algorithms can be broadly grouped according to the following criteria [41]:

1. Label information. Supervised algorithms take in account the class information, while unsupervised algorithms do not, the training features as belonging to a same class.
2. Search strategy. Filter algorithms (also known as classifier-independent) are based on a two-step "ranking and selecting" criterium: scalar features are first ranked according to a proper criterion; then only the "best" ones are selected. Wrapper methods (also known as classifier-dependent methods) uses the selected classifier, following an "ad hoc" approach: the selected scalar features are those that gives the best classification performance

An example of supervised filter algorithm is the biserial correlation coefficient. Given a training set D composed by N_+ features belonging to the class $+1$ and N_- features belonging to the class -1, the biserial correlation coefficient for the k-th scalar feature x_k is given by [42]:

$$r_k^2 = \frac{\sqrt{N_+ N_-}}{N_+ + N_-} \frac{m(x_k^+) - m(x_k^-)}{s(x_k)} \tag{13}$$

where $m(\cdot)$, $s(\cdot)$ are, respectively, the sample mean and sample standard deviation operators and x_k^+, x_k^- are the subset of x_k belonging to, respectively, the classes $+1$ and -1. The total feature

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

score is obtained summing the m coefficients of each scalar feature x_k. Once sorted the scores r_k^2 in descendent order, the feature selection is made simply by selecting the first scalar features whose summing score get a percentage (e.g. 95%) of the total feature score.

3. Materials and Methods

3.1. Stimuli and experimental paradigm

In the present experiment, we used two sensory categories of stimuli (i.e., visual and auditory), which were extracted from two standardized international archives. Visual stimuli consisted of pictures of 28 faces extracted from the NIMSTIM archive [43], whereas auditory stimuli consisted of 28 sounds chosen from the International Affective Digitized Sounds (IADS) archive [44].

For each sensory category, the stimuli were further extracted according to their arousal value. We selected 14 neutral faces and 14 fearful faces from the NIMSTIM inventory, and 14 low- and 14 high-arousal sounds were selected from the IADS repertoire and balanced by arousal with the NIMSTIM stimuli set. These materials are available at: https://doi.org/10.6084/m9.figshare.6874871.v3 [37]

All participants were presented with two different experimental tasks, which were delivered in separate blocks (see Figure 1). We defined the two condition as Passive and Active Task. Both the block presentation order and the response button were counterbalanced between subjects to avoid possible response biases. The two tasks are described in the Figure 1. The figure illustrates the sequence of events and the temporal trial structure relative to the passive (top) and the active (bottom) tasks, which were delivered in blocks. Within each task, the stimuli were randomly presented and equally distributed according to either sensory category (faces or sounds) and arousal level (high or low)

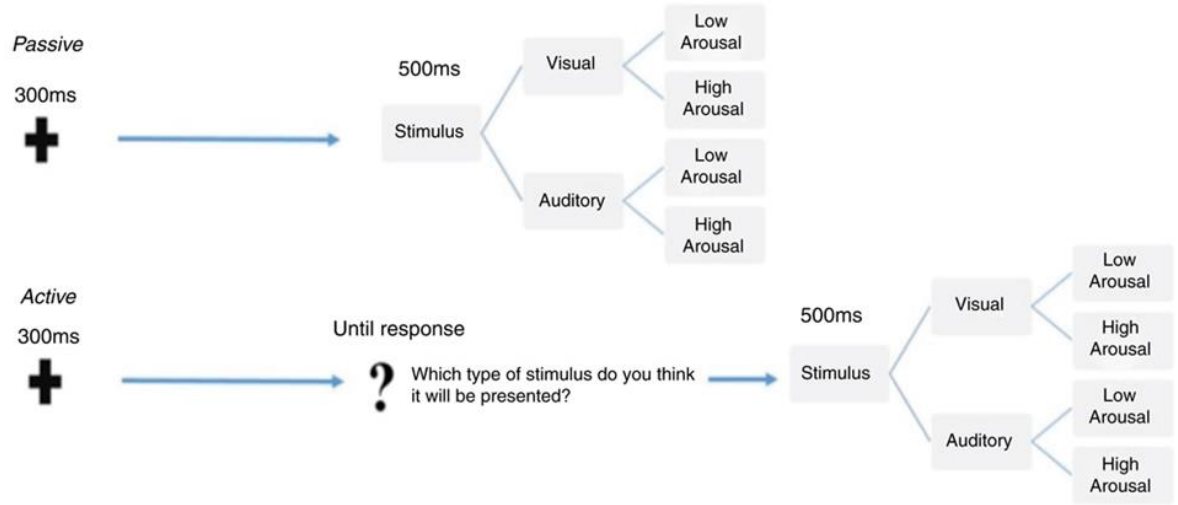


Figure 1. Experimental tasks.

3.1.1 Passive task

As shown in Figure 1, at the beginning of each trial, participants were presented with a warning signal, a fixation cross presented centrally on the screen for 300 ms. After that, a fixed 1000ms blank inter-stimulus interval (ISI) was delivered, followed by a 500ms target stimulus. The target stimulus could be either the picture of a face presented on the centre of the screen or a sound delivered bilaterally through two loudspeakers, with a 50% distribution. Half of the stimuli within each category were low-arousal and the other half were high-arousal, equally distributed. Participants were told that they had to guess which kind of stimulus they would be presented with. No behavioural responses were required until they actually received the stimulus target. At target onset, participants had to discriminate between visual or auditory stimuli by pressing two different buttons on the response box. The response buttons were counterbalanced across participants. After the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

response, the stimulus target disappeared and a blank screen was presented for a jittered duration between 1000 and 1200 ms (inter-trial interval) before the beginning of the next trial.

3.1.2 Active task

In the active task, event sequence and timing were the same as those in the passive task. As illustrated in Figure 1, the only difference in comparison with the passive task was that, after the pre-stimulus ISI, participants were presented with a slide showing a central question mark. They were then asked to make an explicit choice about the sensory category of the upcoming stimulus by pressing the response box. This allowed us to obtain an overt behavioural measure of the anticipation of random events as the percentage of correct responses compared to chance-level for each stimulus category. Immediately after participants' response, stimuli were presented for 500 ms. As with the passive task, the response buttons were counterbalanced across participants.

A total of 200 trials sorted by stimulus category and arousal was presented in each task, for an experiment duration of about 18 minutes. In both tasks, stimuli presentation was fully randomized. Specifically, the trial-type randomization was generated online during the ISI by using a true random number generator (TrueRNG-2™). The TrueRNG hardware uses the avalanche effect in a semiconductor junction to generate true random numbers. Randomization via an external TrueRNG device does not rely on seed-based randomization algorithms, but on current fluctuations within the device, assuring a true random distribution. The RNG was interfaced with the stimuli presentation software E-Prime™ 2.0.8.90.

3.2 EEG recording

During the entire experiment, the EEG signal was continuously recorded using a Geodesic high-density EEG system (EGI GES-300) through a pre-cabled 128-channel HydroCel Geodesic Sensor Net (HCGSN-128) referenced to the vertex (CZ), with a sampling rate of 500 Hz. The impedance was kept below 60kΩ for each sensor. To reduce the presence of EOG artefacts, subjects were instructed to limit both eye blinks and eye movements, as much as possible.

3.3 EEG preprocessing

The continuous EEG signal was off-line band-pass filtered (0.1-45Hz) using a Hamming-windowed sinc FIR (finite impulse response) filter (order = 16500) and then downsampled at 250 Hz. The EEG was epoched starting from 200 ms before the cue onset and ending at the stimulus onset. The initial epochs were 1300 ms long from the cue onset, including 300 ms of cue/fixation cross presentation and 1000 ms of interstimulus interval (ISI).

All epochs were visually inspected to remove bad channels and rare artefacts. Artefact-reduced data were then subjected to the ICA (independent component analysis) [45]. All independent components were visually inspected, and those related to eye blinks, eye movements, and muscle artefacts, according to their morphology and scalp distribution, were discarded. The remaining components were back-projected to the original electrode space to obtain cleaner EEG epochs.

The remaining ICA-cleaned epochs that still contained excessive noise or drift (±100 μV at any electrode) were rejected and the removed bad channels were reconstructed. Data were, then, re-referenced to the CAR (common average reference) and the epochs were baseline-corrected by subtracting the mean signal amplitude in the pre-stimulus interval. From the original 1300 ms long epochs, final epochs were obtained only from the 1000 ms long ISI.

3.4. Static Spectral Features

From each epoch and each channel k, the PSD (Power Spectral Density) was estimate by a Welch's periodogram using a 250 points long Hamming's windows with 50% overlapping. PSD was first log-transformed to compensate the skewness of power values [46], then the spectral bins corresponding to alpha, beta and theta bands – defined, respectively, as $13{\sim}30Hz$, $6{\sim}13Hz$ and $4{\sim}6Hz$ [47] – were summed together. Finally, alpha, beta and theta total powers were computed as:

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

$$\begin{aligned}\beta_{tot}^{k} &= \sum_{i\in[13;30]} PSD^{k}(i)\\ \alpha_{tot}^{k} &= \sum_{i\in[6;13]} PSD^{k}(i)\\ \theta_{tot}^{k} &= \sum_{i\in[4;6]} PSD^{k}(i)\end{aligned} \tag{14}$$

As a measure of the emotional arousal, we computed ratio between beta and alpha total powers $\beta_{tot}^{k}/\alpha_{tot}^{k}$ [48], while to measure the cognitive arousal we computed the ratio between beta and theta total powers $\theta_{tot}^{k}/\beta_{tot}^{k}$ [49].

For each epoch, the feature (with a dimensionality of 256) was obtained concatenating the beta-over-alpha and beta-over-theta ratio of all the channel:

$$v \doteq \left[\frac{\beta_{tot}^{1}}{\alpha_{tot}^{1}}, \frac{\theta_{tot}^{1}}{\beta_{tot}^{1}}, \frac{\beta_{tot}^{2}}{\alpha_{tot}^{2}}, \frac{\theta_{tot}^{2}}{\beta_{tot}^{2}}, \ldots, \frac{\beta_{tot}^{128}}{\alpha_{tot}^{128}}, \frac{\theta_{tot}^{128}}{\beta_{tot}^{128}}\right] \tag{15}$$

3.5. Static Temporal features

It has been previously showed that arousal level (high or low) can be estimated from the Contingent Negative Variation potentials [37]. The feature extraction procedure, therefore, follows the classical approach for the Event-Related Potentials [50]. Each epoch from each channel was first band pass filtered ($0.05\sim10Hz$) using a zero-phase 2nd order Butterworth filter and decimated to a sample frequency of $20Hz$. EEG signal was thus normalized (i.e. z-scored) according to the temporal mean and the temporal standard deviation:

$$x_i(t_k) = (\tilde{x}_i(t_k) - m_i)/s_i$$

where $\tilde{x}_i(t_k)$ is the raw signal from i-th channel at time point t_k, m_i and s_i are, respectively, the temporal mean and the temporal standard deviation of the i-th channel. For each epoch, the feature (with a dimensionality of 2560) was obtained concatenating all normalized signal from each channel:

$$v \doteq [x_1(t_1), x_1(t_2), \ldots, x_1(t_{20}), \ldots, x_{128}(t_1), x_{128}(t_2), \ldots, x_{128}(t_{20})] \tag{16}$$

3.6. Dynamic Features

Each epoch was partitioned into 125 temporal segments, 500 ms long and shifted by 1/250 s (1 sample). Within each time segment, we extracted the dynamic spectral and temporal features, following the same approaches described in, respectively, sect. 3.4 and 3.5. Dynamic temporal features had a dimensionality of 1280, corresponding to $0.5 \times 20 = 10$ samples per channel. Dynamic spectral features had the same dimensionality of their static counterparts (256), but the Welch's periodogram was computed using a 16 points long Hamming's window (zero-padded to 250 points) with 50% of overlapping.

3.7. Feature reduction and Classification

The extracted features (both static and dynamic) were grouped according to the stimulus type (sound or image) and the task (active or passive), in order to classify the group-related arousal level (high or low). A total of 4 binary classification problems (high arousal VS low arousal) were performed: active image (Ac_Im), active sound (Ac_So), passive image (Ps_Im) and passive sound (Ps_So).

Static features features were reduced by means of the biserial correlation coefficient r^2 with the threshold set at 90% of the total feature score. In order to identify the discriminative power of each EEG channel, a series of scalp plots (one for each feature type and each group) of the coefficients were drawn. Since each channel is associated with $N > 1$ features (as well as N r^2 coefficients), the coefficients (one coefficient for each channel) are calculated as mean value. In other words, spectral and temporal features had, respectively, 2 and 20 scalar features for each EEG channel: to compute their scalp plots, we averaged, respectively 2 and 20 r^2 coefficients of each channel. To enhance the

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

visualization of the plots, the coefficients were finally normalized to the total score and expressed as percentage.

Each classification problem was addressed by mean of 3 classifiers: LDA with pseudo-inverse covariance matrix, soft-margin SVM with penalty parameter $C = 1$ and RBF kernel, kNN with euclidean distance and k=1. Additionally, a random classifier, giving a uniform pseudo-random class ($\Pr\{\text{HA}\} = \Pr\{\text{LA}\} = 0.5$), served as a benchmark [35]. The accuracy of the classifiers was measured repeating for 10 times a 10-folds crossvalidation scheme. The feature selection was computed within each crossvalidation step, to avoid overfitting and reduce biased results [42].

For each group (Ac_Im, Ac_So, Ps_Im, Ps_So) and each feature type (static spectral, static temporal), the classification produced a 10×4 matrix containing the mean accuracies (one for each of the 10-folds crossvalidation repetitions) of each classifier.

Dynamic features were reduced and classified similarly to the static ones. For each temporal segment, the associated features were reduced by means of the biserial correlation coefficient (threshold at 90%) and the classifiers (SVM, kNN, LDA and random) were evaluated using a 10-folds crossvalidation scheme – repeated for 10 times.

For each group, each feature type (dynamic spectral, dynamic temporal), each temporal segment and each classifier, the classification produced 10 sequences of mean accuracies $\{ACC_i\}_{i=1}^{125}$ – one for each repetition of the 10-fold crossvalidation scheme.

3.7. Statistical Analysis

The results of the static classifications were compared against the benchmark classifier by means of 2-samples t-test (right tail).

The results of dynamic classifications (i.e. based on dynamic spectral or dynamic temporal features) were compared following a segment-by-segment approach. For each group, the accuracy sequences of the dynamic classifiers (SVM, kNN and LDA) were compared with the benchmark accuracy sequence. Each sample ACC_i^k, with $k = \{\text{SVM}, \text{kNN}, \text{LDA}\}$, was tested against ACC_i^{Random} by means of 2-sample t-tests (right tail). The corresponding p-value sequences $\{p_i^k\}_{i=1}^{125}$ were Bonferroni-Holm corrected for multiple comparisons. Finally, the best accuracy point was detected as the left extreme of the temporal window corresponding to the highest significant accuracy.

4. Results

4.1. Static Features

In Figures 1-2 are shown the scalp distributions of r^2 coefficients for each binary static classification problem, grouped for feature (spectral, temporal) and groups (Ps_Im, Ps_So, Ac_Im, Ac_So).

The temporal feature gave the most consistent topographical pattern, showing that the regions that best differentiate between high vs low stimuli (auditory and visual) were located over the central-parietal electrodes, whereas a more diffuse pattern in the scalp topography emerged for the spectral features.

In Figures 3-4 are shown the box plots of the accuracies of static temporal and spectral classifications, grouped for condition. From left to right: LDA, SVM, kNN and random. Note that SVM accuracies (the 2nd boxplot from the left) are always shown as a line because its accuracies were constant within each crossvalidation step (see also Tables 1-2).

Note that all the accuracies refer to the same static classification problem (High arousal VS Low arousal), performed using different classifiers (SVM, LDA, kNN) and features (spectral, temporal), on different groups (Ps_Im, Ps_So, Ac_Im, Ac_So).

Using spectral features, in only 2 groups some classifiers showed an accuracy greater than the benchmark. In Ac_So group, $ACC_{SVM} = 50.9\%$ ($t(18)=2.371$, $p=0.015$) and $ACC_{kNN} = 50.9\%$ ($t(18)=1.828$, $p=0.042$), while Ps_Im, $ACC_{LDA} = 51.4\%$ ($t(18)=4.667$, $p<0.001$) and $ACC_{SVM} = 51.8\%$ ($t(18)=9.513$, $p<0.001$).

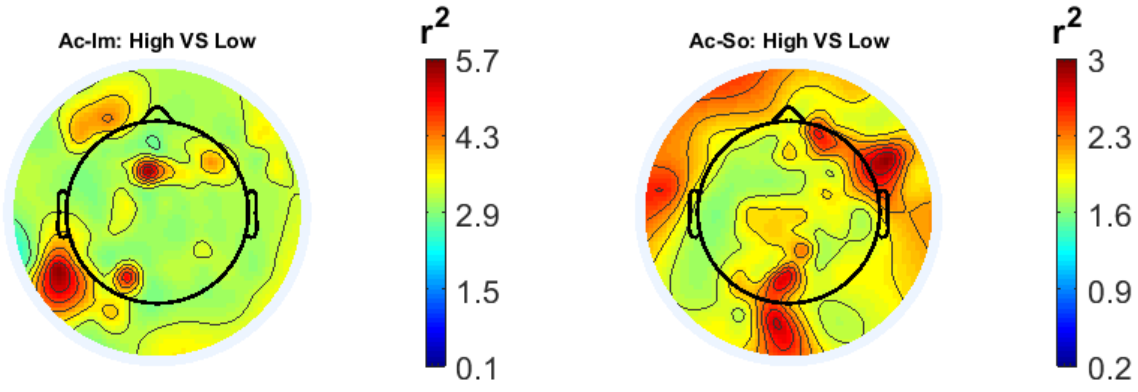
Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

Using temporal features, in all the groups some classifiers showed accuracy greater than the benchmark. In Ac_Im group, $ACC_{SVM} = 50\%$ (t(18)=5.099, p<0.001), in Ac_So $ACC_{SVM} = 50.9\%$ (t(18)=2.907, p=0.005) and $ACC_{kNN} = 51\%$ (t(18)=2.793, p=0.006), in Ps_Im $ACC_{SVM} = 50\%$ (t(18)=9.493, p<0.001) and in Ps_So $AAC_{SVM} = 50.4\%$ (t(18)=9.493, p<0.001).

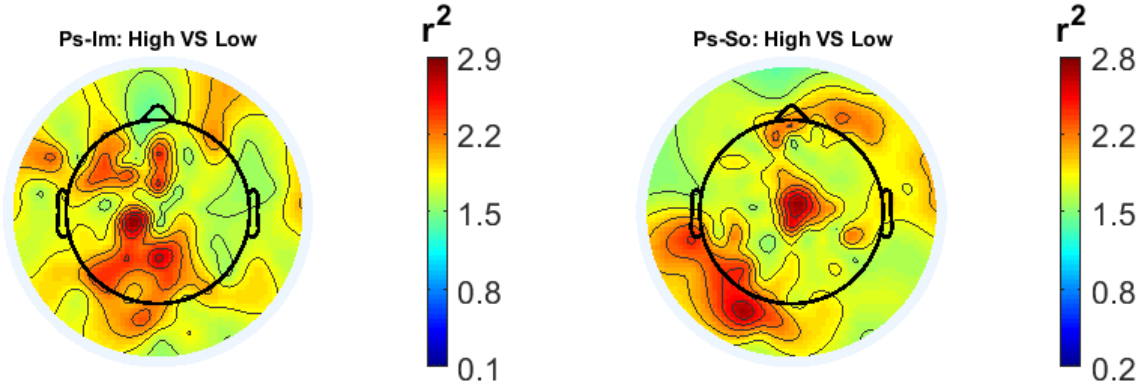
The following Table 1 report the accuracies for static features, ordered in descendent order and grouped for classifier, feature and group.

Classifier	Accuracy	Feature	Group
SVM	51.80%	Spectral	Ps_Im
LDA	51.40%	Spectral	Ps_Im
kNN	51%	Temporal	Ac_So
kNN	50.90%	Spectral	Ac_So
SVM	50.90%	Spectral	Ac_So
SVM	50.90%	Temporal	Ac_So
SVM	50.40%	Temporal	Ps_So
SVM	50.00%	Temporal	Ac_Im
SVM	50.00%	Temporal	Ps_Im

Table 1. Static features. Ordered accuracies grouped for classifier, feature and group.



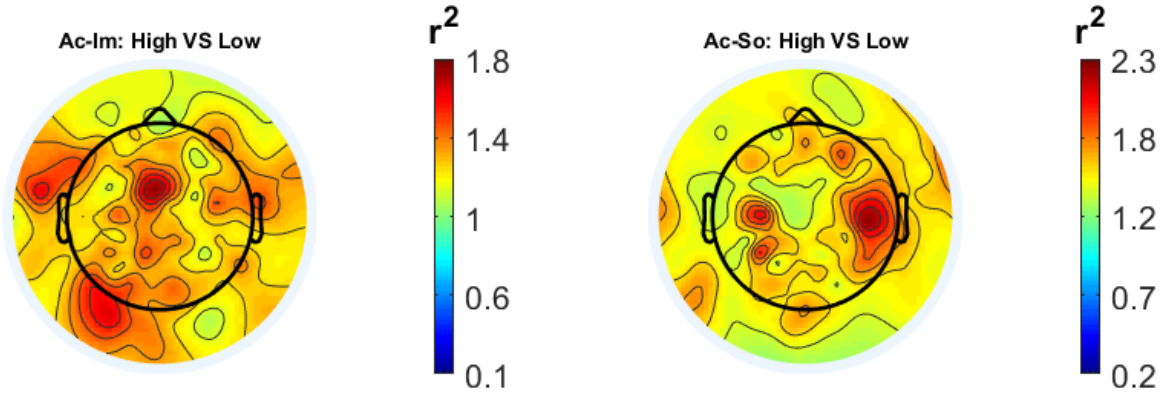
(a)



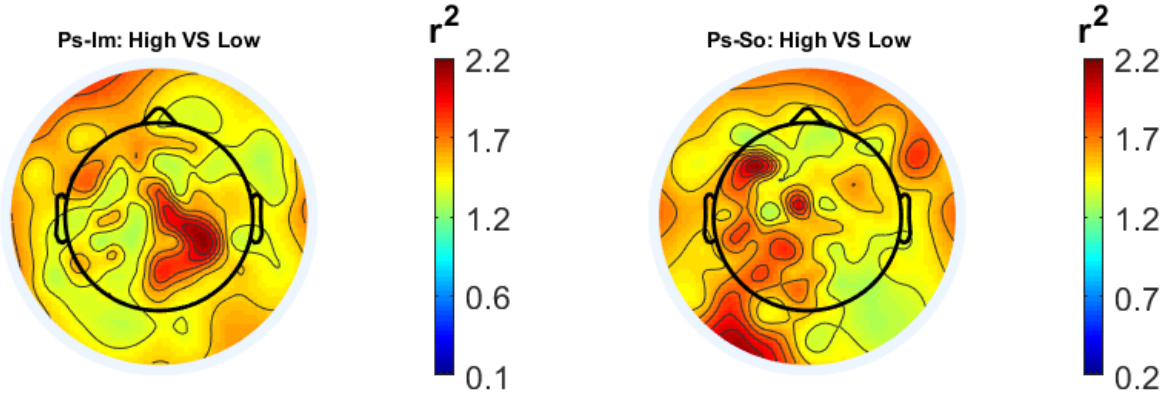
(b)

Figure 1. Spectral features. Scalp distribution of the r^2 coefficients (normalized to the total score and expressed as percentage) grouped for tasks and stimulus type. (**a**) Active task: left Image, right Sound; (**b**) Passive task: left Image, right Sound.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1



(a)



(b)

Figure 2. Temporal features. Scalp distribution of the r^2 coefficients (normalized to the total score and expressed as percentage), grouped for tasks and stimulus type. (a) Active task: left Image, right Sound; (b) Passive task: left Image, right Sound.

Group	LDA	SVM	kNN	Random
Ac_Im	M=0.496, SD=0.007	M=0.510, SD=0.000	M=0.500, SD=0.010	M=0.505, SD=0.011
Ac_So	M=0.492, SD=0.004	M=0.509, SD=0.000	M=0.509, SD=0.007	M=0.503, SD=0.009
Ps_Im	M=0.514, SD=0.010	M=0.518, SD=0.000	M=0.496, SD=0.010	M=0.495, SD=0.008
Ps_So	M=0.488, SD=0.005	M=0.504, SD=0.000	M=0.493, SD=0.007	M=0.503, SD=0.013

Table 2. Mean (M) and standard deviations (SD) of the accuracies of the static spectral classifications. Active Image (Ac_Im), Active Sound (Ac_So), Passive Image (Ps_Im) and Passive Sound (Ps_So).

Group	LDA	SVM	kNN	Random
Ac_Im	M=0.492, SD=0.010	M=0.510, SD=0.000	M=0.500, SD=0.008	M=0.498, SD=0.007
Ac_So	M=0.501, SD=0.007	M=0.509, SD=0.000	M=0.510, SD=0.006	M=0.498, SD=0.012
Ps_Im	M=0.500, SD=0.012	M=0.518, SD=0.000	M=0.492, SD=0.005	M=0.499, SD=0.006
Ps_So	M=0.499, SD=0.008	M=0.504, SD=0.000	M=0.492, SD=0.006	M=0.498, SD=0.008

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

Table 3. Mean (M) and standard deviations (SD) of the accuracies of the static temporal classifications. Active Image (Ac_Im), Active Sound (Ac_So), Passive Image (Ps_Im) and Passive Sound (Ps_So).

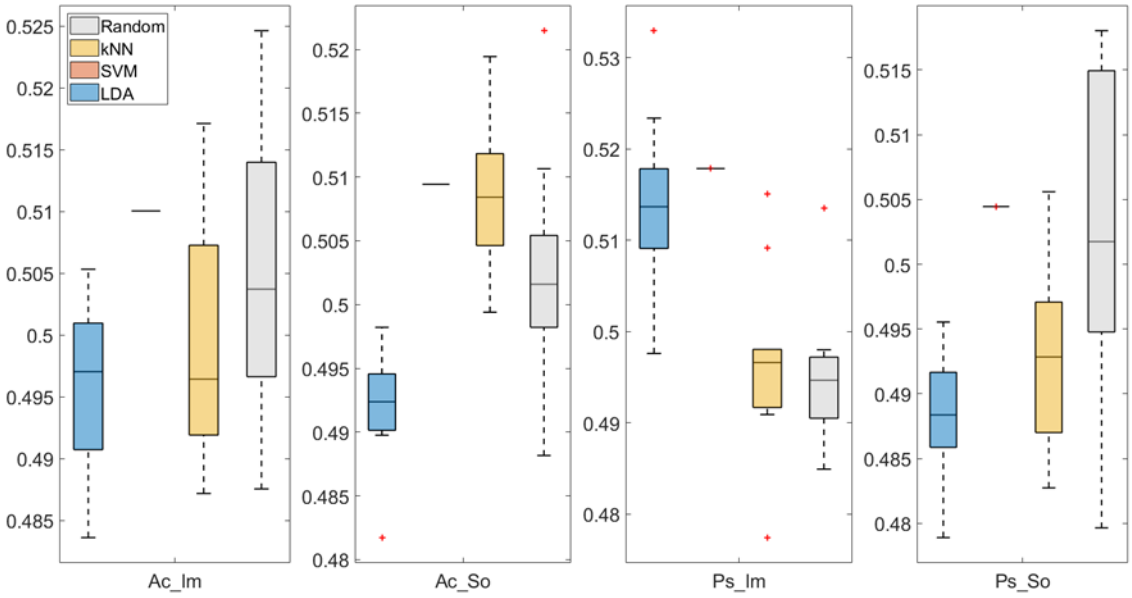


Figure 3. Box-plots of the accuracies of the static spectral classifications. From left: Active Image (Ac_Im), Active Sound (Ac_So), Passive Image (Ps_Im) and Passive Sound (Ps_So).

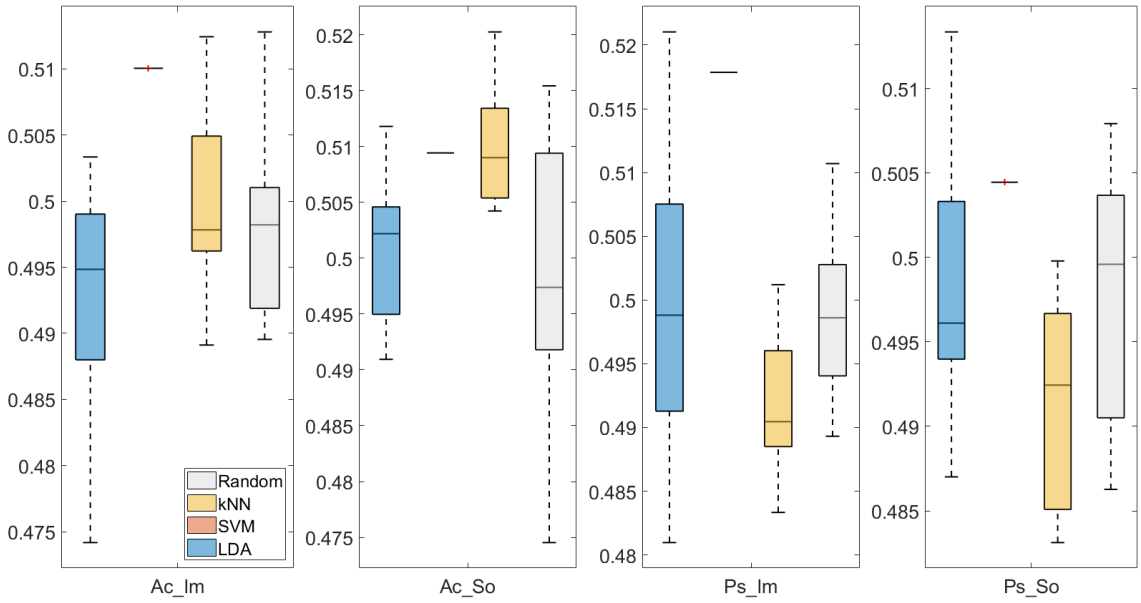


Figure 4. Box-plots of the accuracies of the static temporal classifications From left: Active Image (Ac_Im), Active Sound (Ac_So), Passive Image (Ps_Im) and Passive Sound (Ps_So).

4.2. Dynamic features

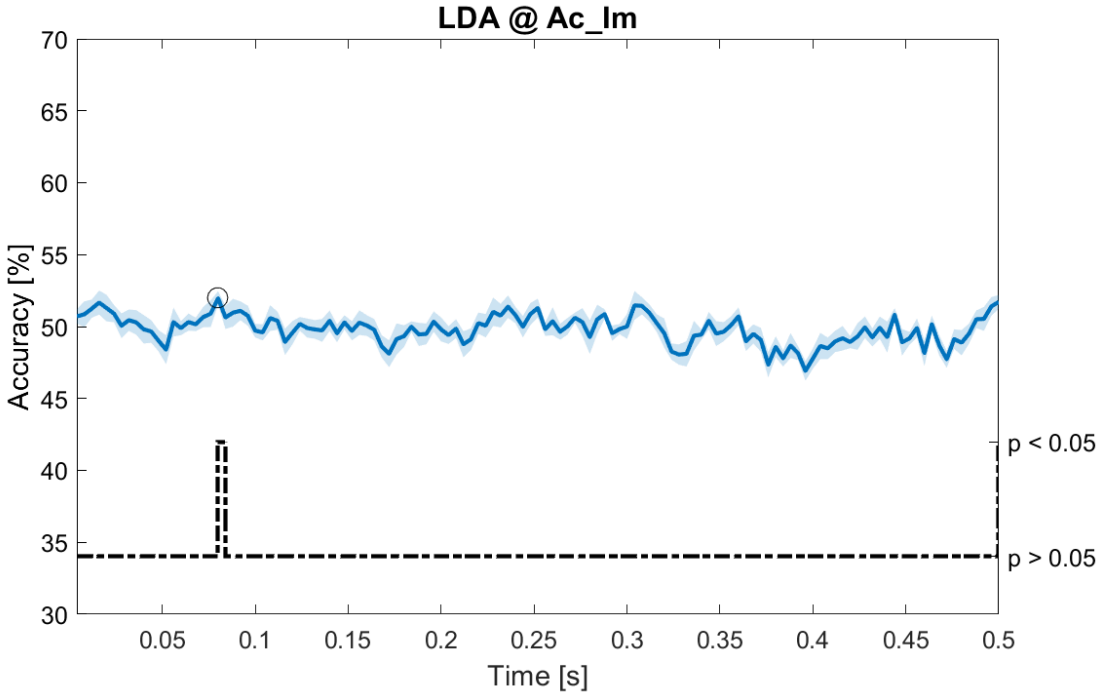
In the following Figures 5-11 are shown the results of the significant dynamic classifications. In the upper plot is shown the mean (bold line) and the standard deviation (shaded) of the accuracy sequence. In the lower plot (black dashed line) it is shown the Bonferroni-Holm corrected p-values sequence, discretized (as a stair graph) as significant ($p<0.05$) or not-significant ($p>0.05$).

Note that all the accuracy plots refer to the same dynamic classification problem (High arousal VS Low arousal), performed using different classifiers (SVM, LDA, kNN) and features (spectral, temporal), on different groups (Ps_Im, Ps_So, Ac_Im, Ac_So).

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

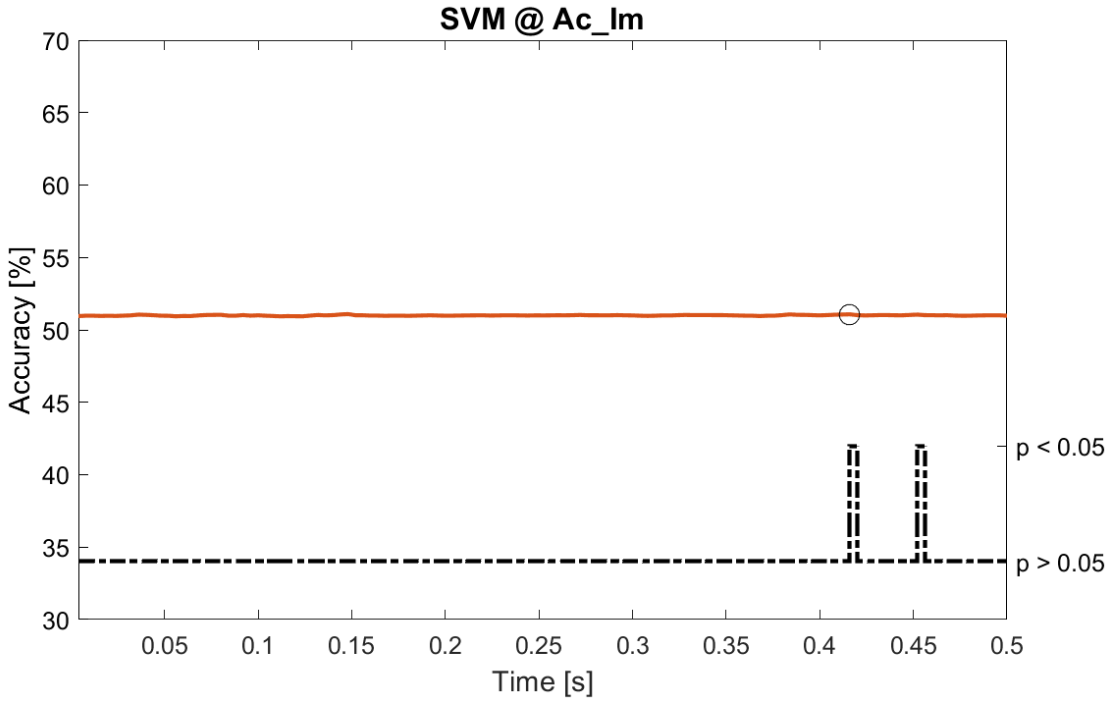
Using spectral features, in all the groups some classifiers showed an accuracy greater than the benchmark. In Ac_Im group, $ACC_{LDA} = 51.97\%\ @t = 0.080s$ (t(18)=6.291, p<0.001) and $ACC_{SVM} = 51.07\%\ @t = 0.416s$ t(18)=6.531, p<0.001). In Ac_So group, $ACC_{LDA} = 53.04\%\ @t = 0.332s$ (t(18)=8.583, p<0.001), $ACC_{SVM} = 51.16\%\ @t = 0.146s$ (t(18)=8.612, p<0.001). In Ps_Im group, $ACC_{LDA} = 53.12\%\ @t = 0.156s$ (t(18)=6.372, p=0.000) and $ACC_{SVM} = 51.83\%\ @t = 0.140s$ (t(18)=6.668, p<0.001). In Ps_So, $ACC_{SVM} = 50.62\%\ @t = 0.024s$ (t(18)=5.236, p=0.003) and $ACC_{kNN} = 51.41\%\ @t = 0.476s$ (t(18)=4.307, p=0.026).

Using temporal features, in only 3 groups some classifiers showed an accuracy greater than the benchmark. In Ac_So group, $ACC_{SVM} = 63.80\%\ @t = 0.100s$ (t(18)=6.113, p=0.001). In Ps_Im group, $ACC_{LDA} = 63.68\%\ @t = 0.024s$ (t(18)=12.108, p<0.001) and $ACC_{SVM} = 51.43\%\ @t = 0.084s$ (t(18)=4.881, p=0.008). In Ps_So group, $ACC_{LDA} = 64.30\%\ @t = 0.0276s$ (t(18)=11.092, p<0.001) and $ACC_{kNN} = 63.70\%\ @t = 0.480s$ (t(18)=16.621, p<0.001).



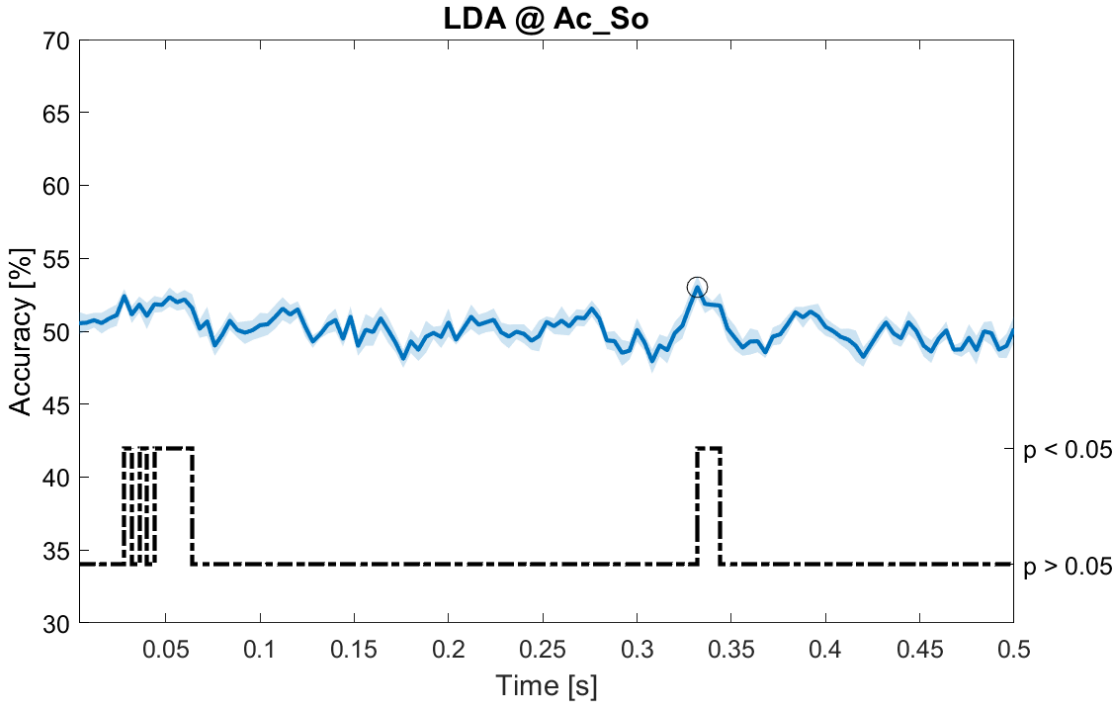
(a)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1



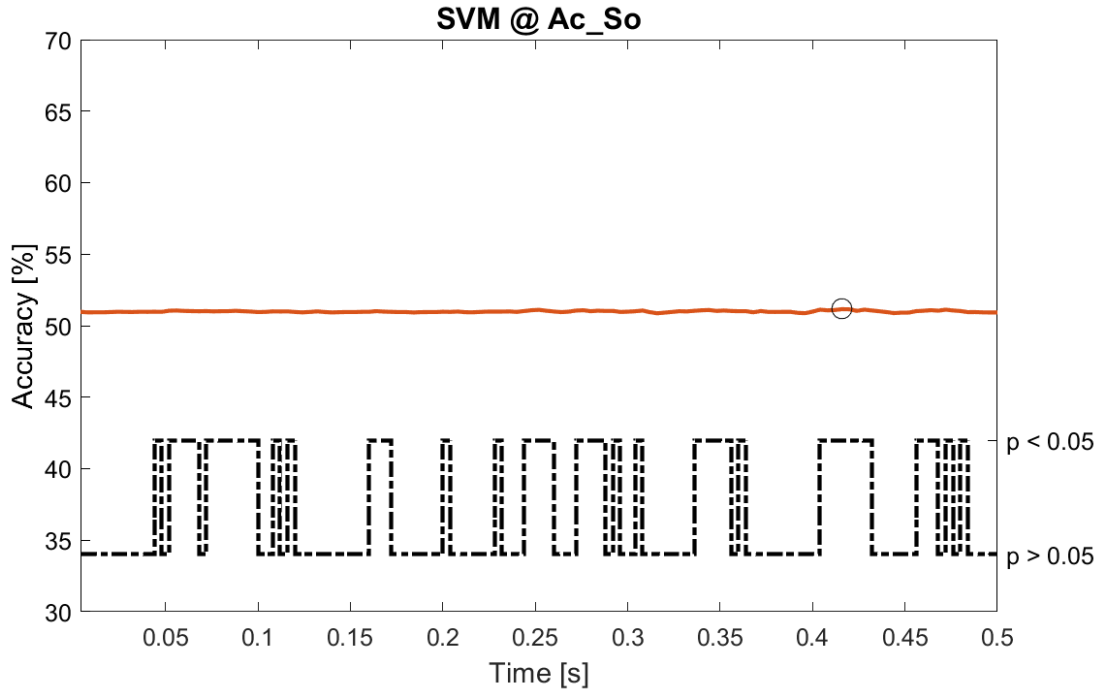
(b)

Figure 5. Spectral dynamic features. Accuracy (mean value, coloured line; standard deviation, shaded line) and p-values (black dotted line) in Ac_Im group for LDA (a) and SVM (b) classifiers.



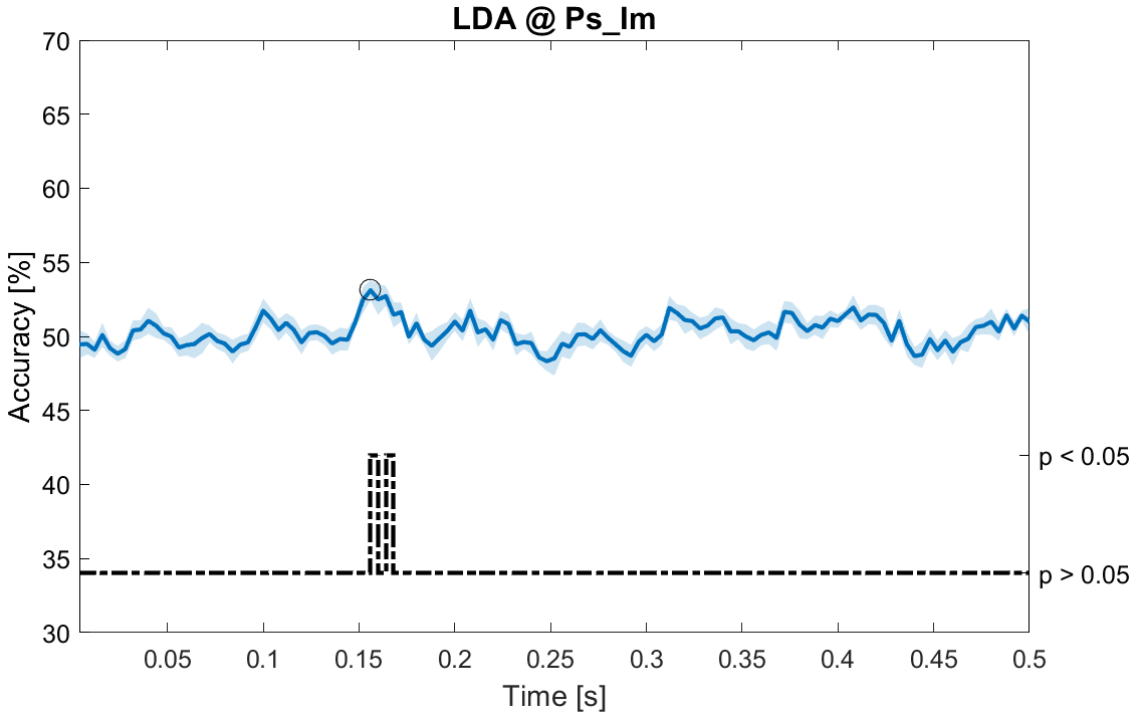
(a)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1



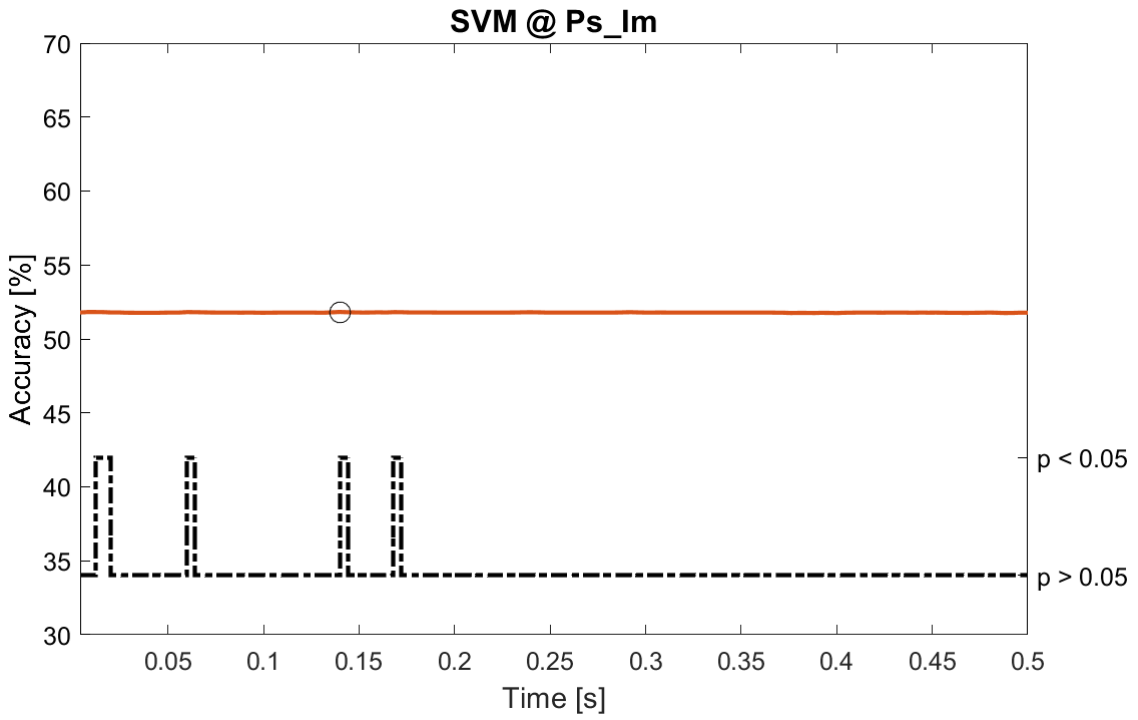
(b)

Figure 6. Spectral dynamic features. Accuracy (mean value, coloured line; standard deviation, shaded line) and p-values (black dotted line) in Ac_So group for LDA (a) and SVM (b) classifiers.



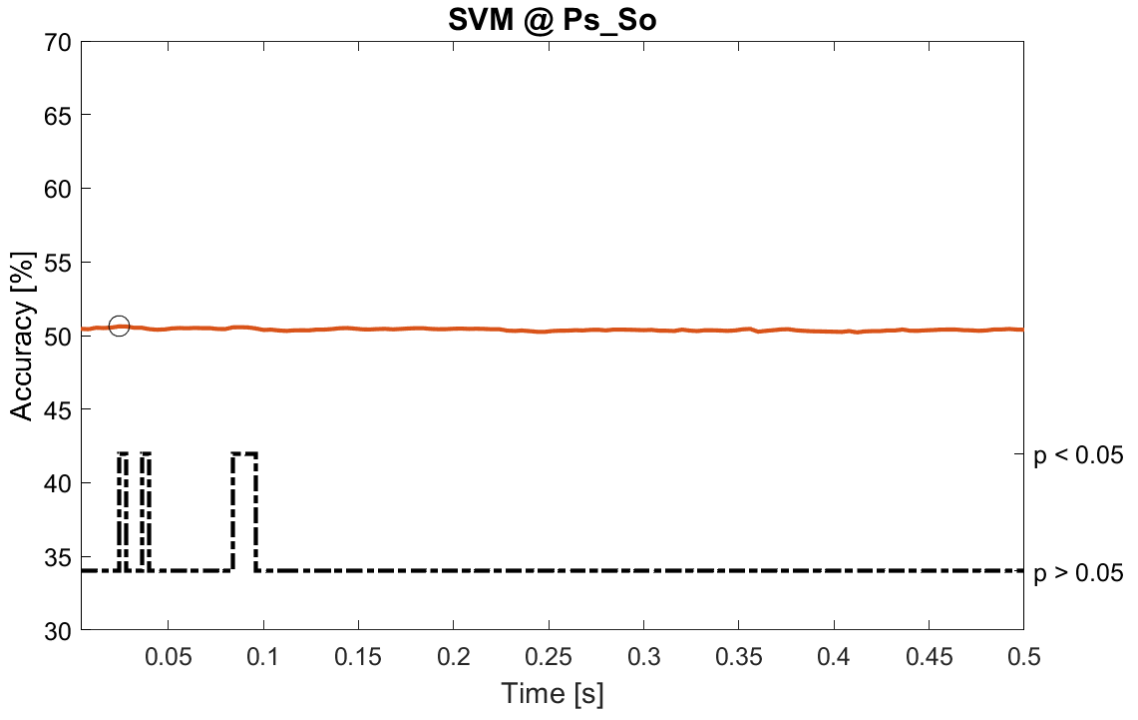
(a)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1



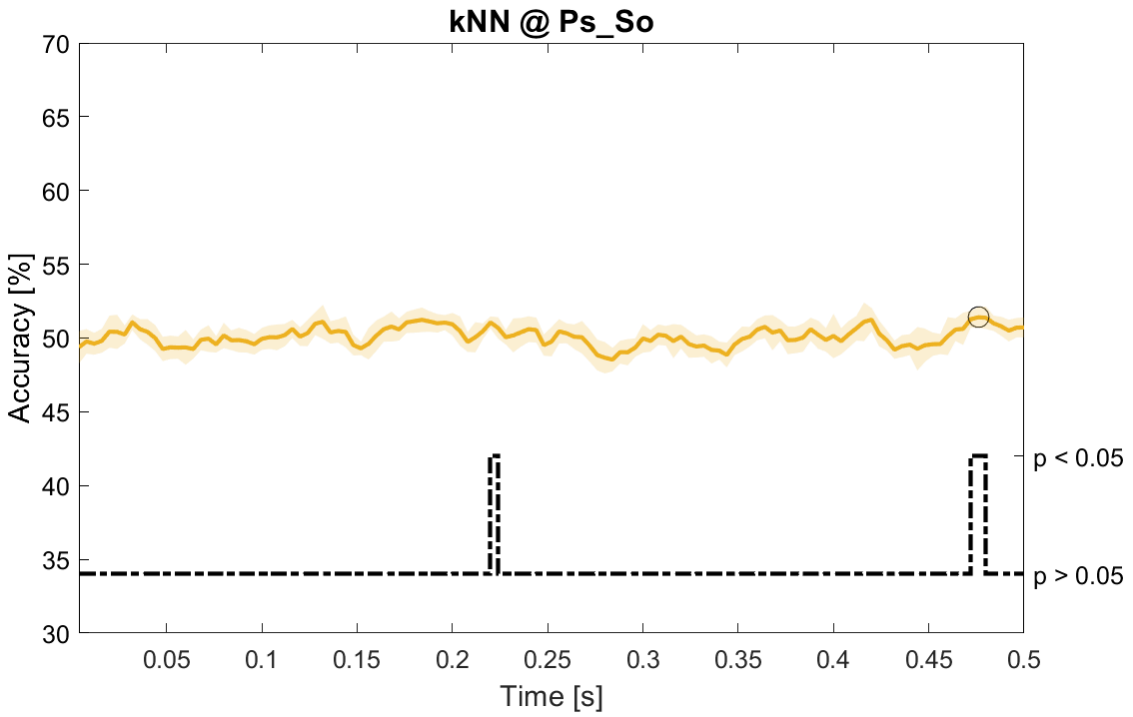
(b)

Figure 7. Spectral dynamic features. Accuracy (mean value, coloured line; standard deviation, shaded line) and p-values (black dotted line) in Ps_Im group for LDA (a) and SVM (b) classifiers.



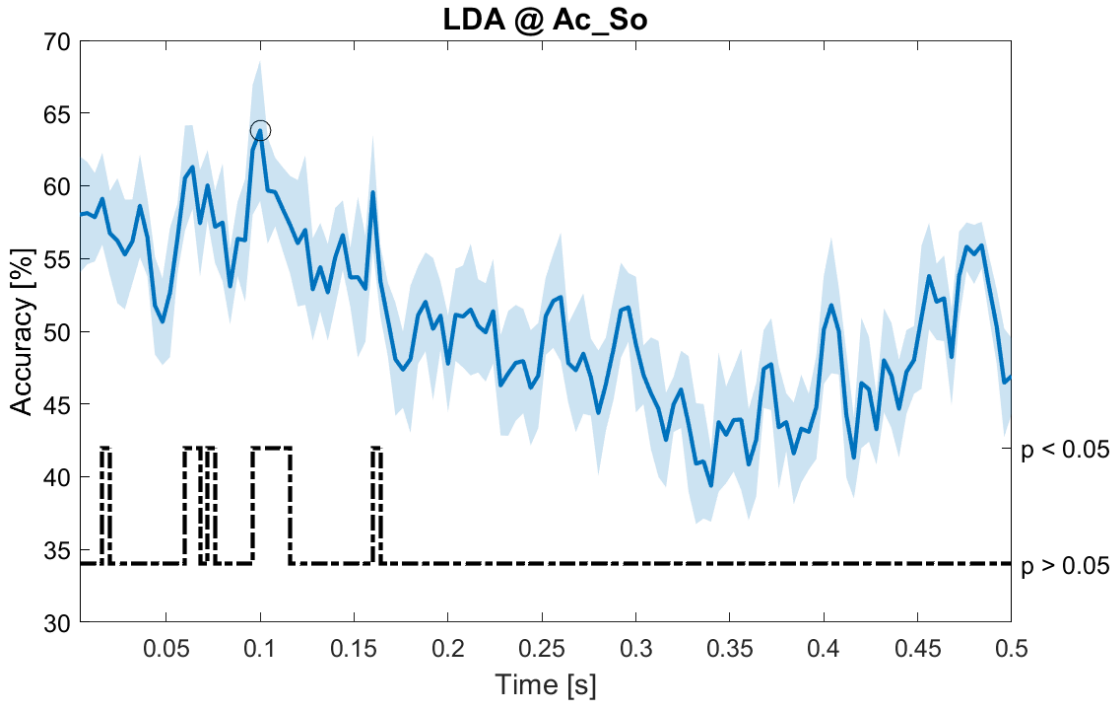
(a)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1



(b)

Figure 8. Spectral dynamic features. Accuracy (mean value, coloured line; standard deviation, shaded line) and p-values (black dotted line) in Ac_So group for SVM (a) and kNN (b) classifiers.



Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1

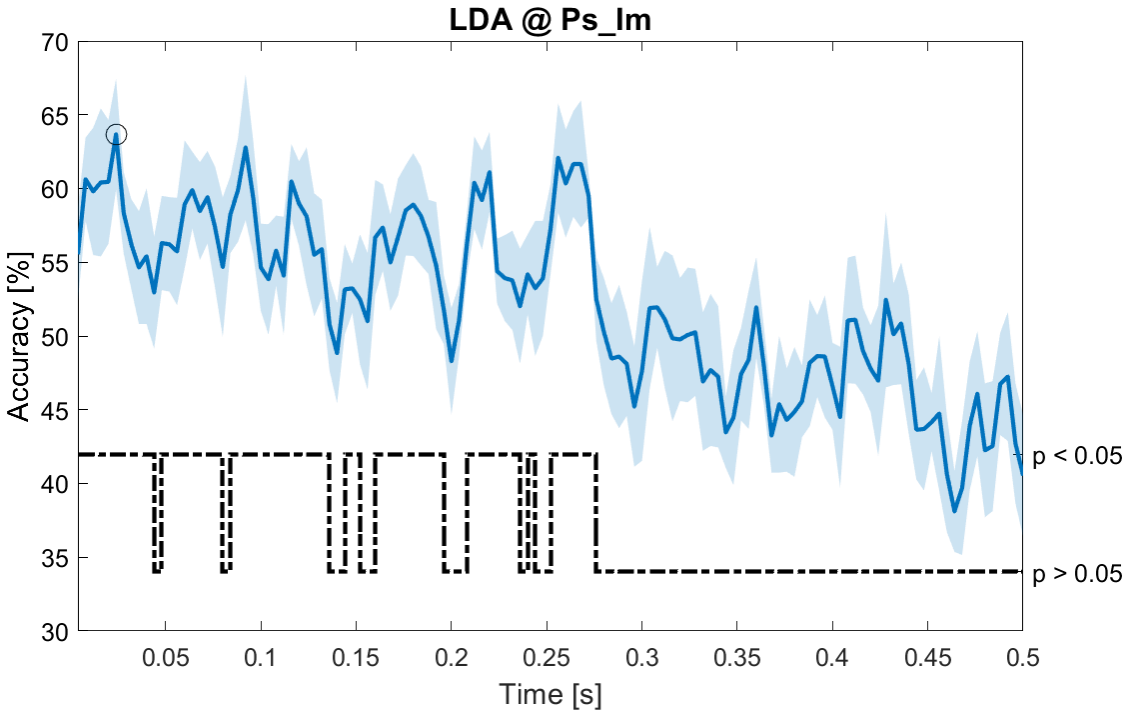
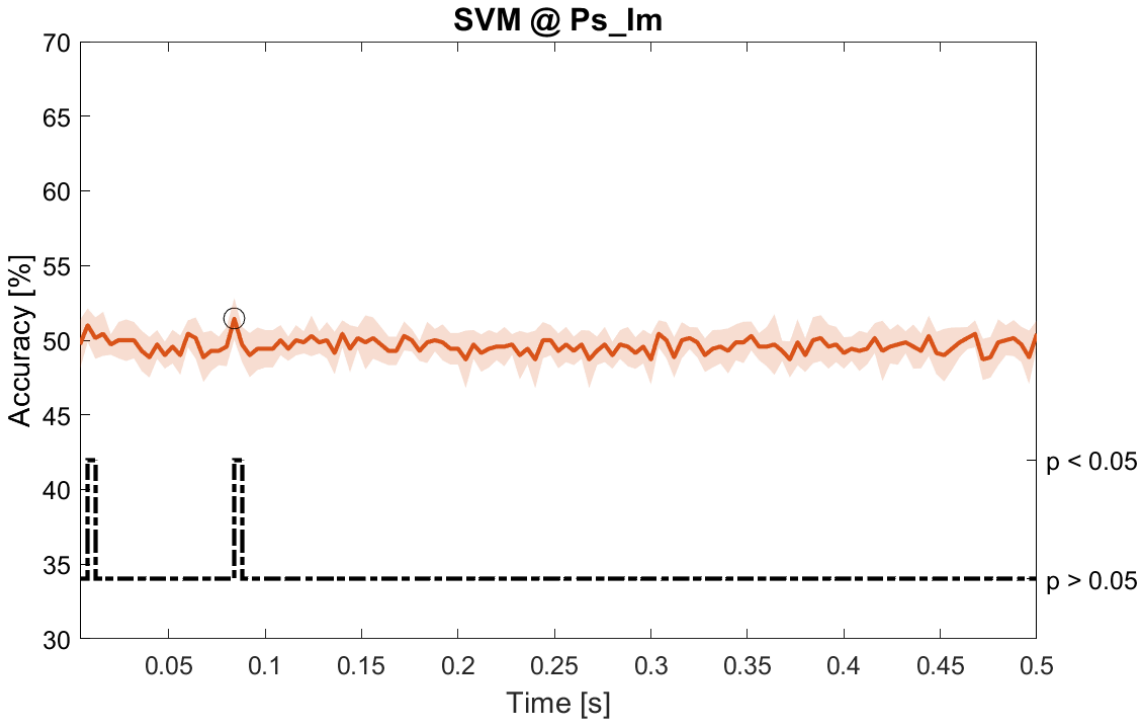


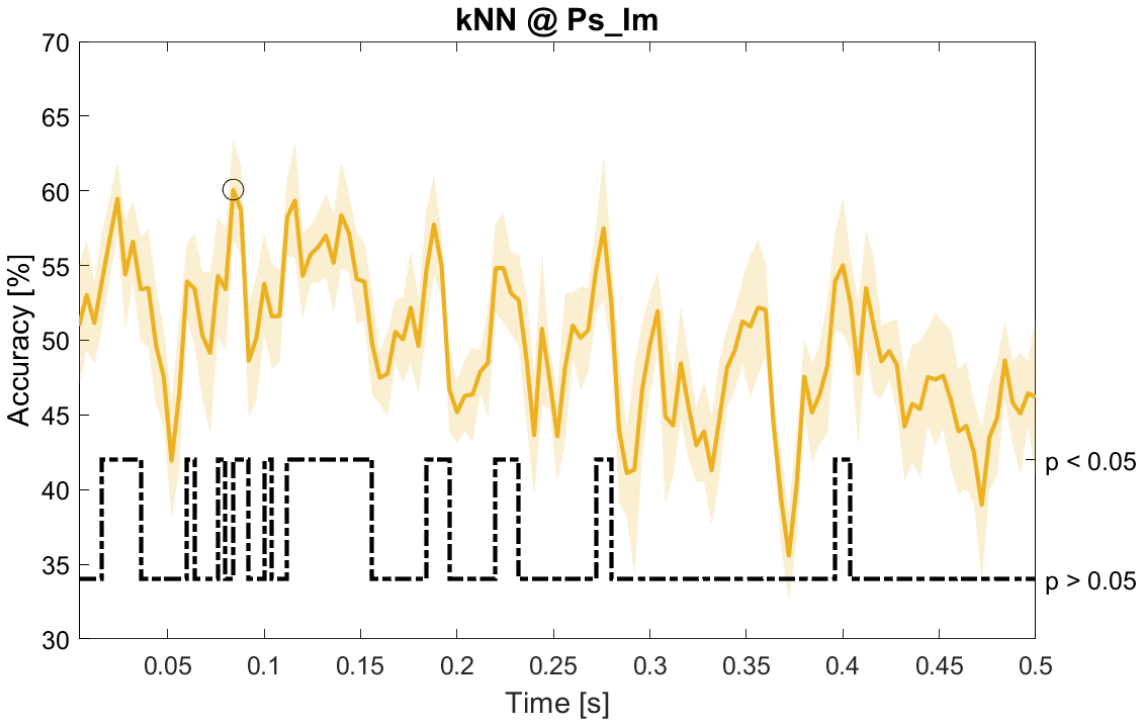
Figure 9. Temporal dynamic features. Accuracy (mean value, coloured line; standard deviation, shaded line) and p-values (black dotted line) in Ac_So group for LDA classifier.

(a)



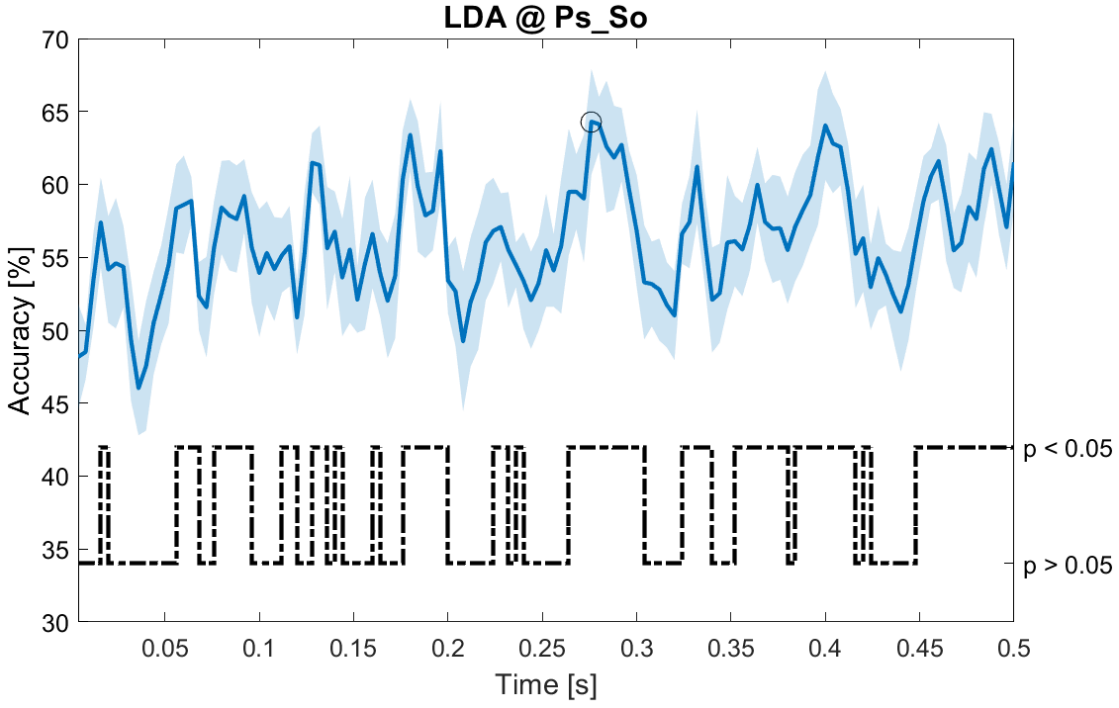
(b)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1



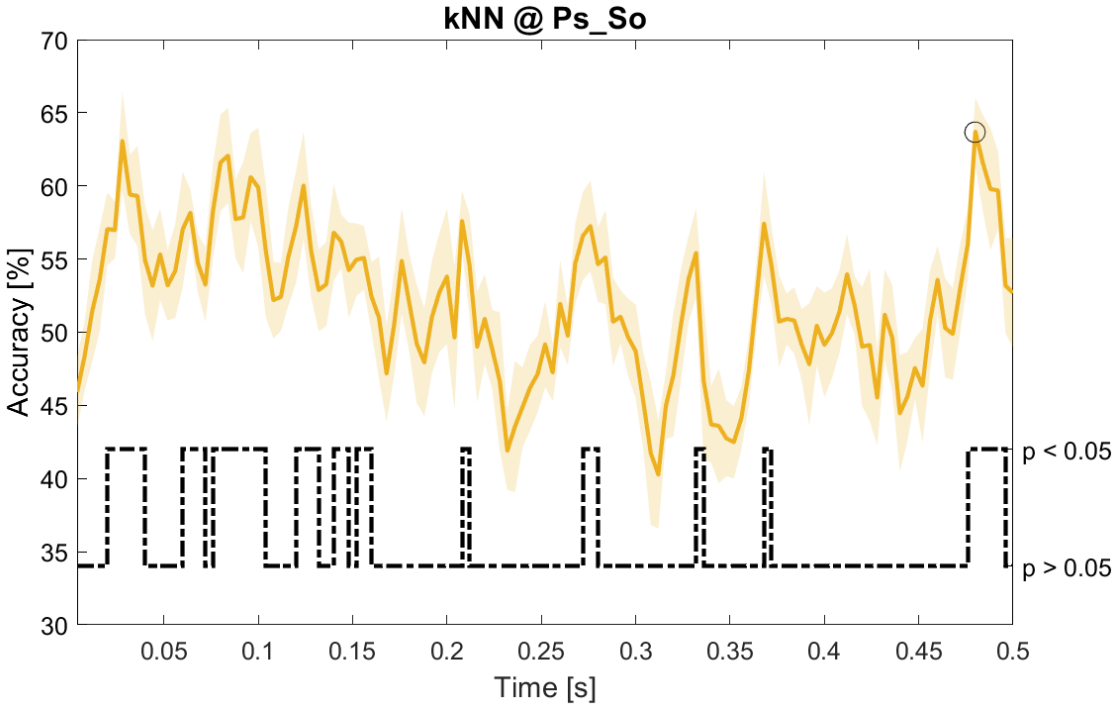
(c)

Figure 10. Temporal dynamic features. Accuracy (mean value, coloured line; standard deviation, shaded line) and p-values (black dotted line) in Ps_Im group for LDA (a), SVM (b) and kNN (c) classifiers.



(a)

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 25 December 2019 doi:10.20944/preprints201912.0340.v1



(b)

Figure 11. Temporal dynamic features. Accuracy (mean value, coloured line; standard deviation, shaded line) and p-values (black dotted line) in Ps_So group for LDA (a) and kNN (b) classifiers.

The following Table 4 reports the accuracies for dynamic features, ordered in descendent order and grouped for classifier, feature group and time.

Classifier	Accuracy	Time [s]	Group	Feature
SVM	63.80%	0.1	Ac_So	Temporal
kNN	63.70%	0.048	Ps_So	Temporal
LDA	63.68%	0.024	Ps_Im	Temporal
LDA	63.30%	0.0276	Ps_So	Temporal
LDA	53.12%	0.156	Ps_Im	Spectral
LDA	53.04%	0.3332	Ac_So	Spectral
LDA	51.97%	0.08	Ac_Im	Spectral
SVM	51.83%	0.14	Ps_Im	Spectral
SVM	51.43%	0.084	Ps_Im	Temporal
kNN	51.41%	0.476	Ps_So	Spectral
SVM	51.16%	0.146	Ac_So	Spectral
SVM	51.07%	0.416	Ac_Im	Spectral
SVM	50.62%	0.024	Ps_So	Spectral

Table 4. Dynamic features. Ordered accuracies grouped for classifier, feature and group.

5. Discussion

The aim of the study was to provide new methodological insights regarding the ML approaches in the classification of the anticipatory emotion-related EEG signal, by testing the performance of different classifiers on different features.

From the ISIs (Inter Stimulus Intervals, i.e. the 1000 ms long window preceding each stimulus onset) we extracted two kind of "static" features, namely spectral and temporal, the most commonly used features in the field emotion recognition [19,20]. As spectral features we used the beta-over-

alpha and the beta-over-theta ratio, whereas for the temporal feature we concatenated the decimated EEG values

Additionally, we extracted the temporal sequences both static spectral and temporal features, using a 500 ms long window moving along the ISI to build, respectively, "dynamic" spectral and temporal features. This step is crucial for our work since, considering the temporal resolution of the EEG, an efficient classification should take in account the temporal dimension, to provide information about when the difference between two condition are maximally expressed and therefore classified.

We trained and tested three different classifiers (LDA, SVM, kNN, the most commonly used in the field of emotion recognition [19,20]) using both static and dynamic features comparing their accuracies against a random classifier, that served as benchmark.

Our goal was to identify the best classifier (static VS dynamic) and the best feature type (spectral vs temporal) to classify the arousal level (high VS low) of 56 auditory/visual. The stimuli, extracted from two standardized datasets (NIMSTIM [43] and IADS [44]), for visual and auditory stimuli, respectively) were presented in a randomized order, triggered by a hardware true RNG device.

Considering the number of group (4), the number of classifiers (3) and the number of feature types (2), each classification (static or dynamic) produced a total of 24 accuracies, whose significances were statistically tested (using a 2 samples t-test and the benchmark's accuracies).

Within the 9 significant accuracies obtained using static features, the classifier that obtained the highest number of accuracies was the SVM (6 significant accuracies), followed by kNN (2 significant accuracies) and LDA (1 significant accuracies). The most frequent feature was the temporal (5 significant accuracies). Finally, the best (static) feature-classifier combination was the SVM with spectral features (51.8%), followed by LDA with spectral features (51.4%) and kNN with temporal features (51%).

Within the 13 significant accuracies obtained using dynamic features, the classifier that obtained the highest number of accuracies was the SVM (6 significant accuracies), followed by LDA (4 significant accuracies) and kNN (3 significant accuracies). The most frequent feature was the spectral (8 significant accuracies). Finally, the best (dynamic) feature-classifier combination was the SVM with temporal features (63.8%), followed by kNN with temporal features (63.70%) and LDA with temporal features (63.68%). Spectral features produced only the 5th highest accuracy (53.12% with LDA). The 3 best accuracies were all within the first 100ms of the ISI, albeit a non-significant Spearman's correlation between accuracy and time was observed ($r=-0.308$, $p=0.306$).

Our results show that globally the SVM presents the best accuracy, independently from the feature type (temporal or spectral), but more importantly the combination of SVM with dynamic temporal feature produced the best classification performance. This finding is particularly relevant, considering the application of EEG in cognitive science. In fact, due to its high temporal resolution, EEG is often applied to investigate the timing of neural processes most of the time in relation to behavioural performance.

Our results therefore suggest that, in order to best classify emotions based on the electrophysiological brain activity, temporal dynamic of the EEG signal should be taken in account with a dynamic feature and consequently with a dynamic classifier. In fact, by including also time evolution of the feature in the ML model, it is possible to infer when two different conditions maximally diverge, allowing possible interpretation of the timing of the cognitive processes and the behaviour of the underlying neural substrate.

Finally, the main contribution of our results for the scientific community in ML is that they provide a methodological advancement generally valid both for the investigation of emotion based on ML approach with EEG signal and also for the investigation of preparatory brain activity.

Author Contributions: GM.D., G.M., L.S., and P.T. conceptualized the experimental design; LC developed e-prime code and cured the software interface for the Random Number Generator, M.B. and GM.D. were involved in data curation, developing analytical methodology and perform formal analysis; M.B and GM.D wrote the original draft, P.T. and G.M supervised the writing process.

Funding: This research received no external funding.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Friston, K. A theory of cortical responses. *Philos. Trans. R. Soc. B Biol. Sci.* **2005**, *360*, 815–836.
2. Nobre, A.C. Orienting attention to instants in time. *Neuropsychologia* **2001**, *39*, 1317–1328.
3. Mento, G.; Vallesi, A. Spatiotemporally dissociable neural signatures for generating and updating expectation over time in children: A High Density-ERP study. *Dev. Cogn. Neurosci.* **2016**, *19*, 98–106.
4. Mento, G.; Tarantino, V.; Vallesi, A.; Bisiacchi, P.S. Spatiotemporal neurodynamics underlying internally and externally driven temporal prediction: A high spatial resolution ERP study. *J. Cogn. Neurosci.* **2015**, *27*, 425–439.
5. Barsalou, L.W. Grounded Cognition. *Annu. Rev. Psychol.* **2008**, *59*, 617–645.
6. Barrett, L.F. The theory of constructed emotion: an active inference account of interoception and categorization. *Soc. Cogn. Affect. Neurosci.* **2017**, *12*, 1–23.
7. Bruner, J.S. (Jerome S. *Acts of meaning*; Harvard University Press, 1990; ISBN 9780674003606.
8. Miniussi, C.; Wilding, E.L.; Coull, J.T.; Nobre, A.C. Orienting attention in time. Modulation of brain potentials. *Brain* **1999**, *122*, 1507–1518.
9. Stefanics, G.; Hangya, B.; Hernádi, I.; Winkler, I.; Lakatos, P.; Ulbert, I. Phase entrainment of human delta oscillations can mediate the effects of expectation on reaction speed. *J. Neurosci.* **2010**, *30*, 13578–13585.
10. Denny, B.T.; Ochsner, K.N.; Weber, J.; Wager, T.D. Anticipatory brain activity predicts the success or failure of subsequent emotion regulation. *Soc. Cogn. Affect. Neurosci.* **2014**, *9*, 403–411.
11. Abler, B.; Erk, S.; Herwig, U.; Walter, H. Anticipation of aversive stimuli activates extended amygdala in unipolar depression. *J. Psychiatr. Res.* **2007**, *41*, 511–522.
12. Morinaga, K.; Akiyoshi, J.; Matsushita, H.; Ichioka, S.; Tanaka, Y.; Tsuru, J.; Hanada, H. Anticipatory anxiety-induced changes in human lateral prefrontal cortex activity. *Biol. Psychol.* **2007**, *74*, 34–38.
13. Duma, G.M.; Mento, G.; Manari, T.; Martinelli, M.; Tressoldi, P. Driving with Intuition: A Preregistered Study about the EEG Anticipation of Simulated Random Car Accidents. *PLoS One* **2017**, *12*, e0170370.
14. Radin, D.I.; Vieten, C.; Michel, L.; Delorme, A. Electrocortical activity prior to unpredictable stimuli in meditators and nonmeditators. *Explor. J. Sci. Heal.* **2011**, *7*, 286–299.
15. Mossbridge, J.A.; Tressoldi, P.; Utts, J.; Ives, J.A.; Radin, D.; Jonas, W.B. Predicting theunpredictable: Critical analysis and practical implications of predictive anticipatory activity. *Front. Hum. Neurosci.* **2014**, *8*.
16. Gunes, H.; Pantic, M. Automatic, Dimensional and Continuous Emotion Recognition. *Int. J. Synth. Emot.* **2010**, *1*, 68–99.
17. Shu, L.; Xie, J.; Yang, M.; Li, Z.; Li, Z.; Liao, D.; Xu, X.; Yang, X.; Shu, L.; Xie, J.; et al. A Review of Emotion Recognition Using Physiological Signals. *Sensors* **2018**, *18*, 2074.
18. Calvo, R.A.; D'Mello, S. Affect detection: An interdisciplinary review of models, methods, and their applications. *IEEE Trans. Affect. Comput.* **2010**, *1*, 18–37.
19. Alarcao, S.M.; Fonseca, M.J. Emotions Recognition Using EEG Signals: A Survey. *IEEE Trans. Affect. Comput.* **2017**, *3045*, 1–20.
20. Al-Nafjan, A.; Hosny, M.; Al-Ohali, Y.; Al-Wabil, A.; Al-Nafjan, A.; Hosny, M.; Al-Ohali, Y.; Al-Wabil, A. Review and Classification of Emotion Recognition Based on EEG Brain-Computer Interface System Research: A Systematic Review. *Appl. Sci.* **2017**, *7*, 1239.
21. Lotte, F.; Congedo, M.; Lécuyer, A.; Lamarche, F.; Arnaldi, B. A review of classification algorithms for EEG-based brain-computer interfaces. *J. Neural Eng.* **2007**, *4*, R1–R13.

22. Lin, Y.P.; Wang, C.H.; Wu, T.L.; Jeng, S.K.; Chen, J.H. EEG-based emotion recognition in music listening: A comparison of schemes for multiclass support vector machine. In Proceedings of the ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings; 2009.
23. Koelstra, S.; Yazdani, A.; Soleymani, M.; Mühl, C.; Lee, J.S.; Nijholt, A.; Pun, T.; Ebrahimi, T.; Patras, I. Single trial classification of EEG and peripheral physiological signals for recognition of emotions induced by music videos. In Proceedings of the Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics); 2010.
24. Liu, Y.; Sourina, O. EEG-based valence level recognition for real-time applications. In Proceedings of the Proceedings of the 2012 International Conference on Cyberworlds, Cyberworlds 2012; 2012.
25. Murugappan, M.; Murugappan, S. Human emotion recognition through short time Electroencephalogram (EEG) signals using Fast Fourier Transform (FFT). In Proceedings of the Proceedings - 2013 IEEE 9th International Colloquium on Signal Processing and its Applications, CSPA 2013; 2013.
26. Thammasan, N.; Fukui, K.I.; Numao, M. Application of deep belief networks in EEG-based dynamic music-emotion recognition. In Proceedings of the Proceedings of the International Joint Conference on Neural Networks; 2016.
27. Duda, R.O.; Hart, P.E.; Stork, D.G. *Pattern classification*; 2nd ed.; Wiley, 2000; ISBN 9780471056690.
28. Bishop, C. *Pattern Recognition and Machine Learning*; 1st ed.; Springer, 2006; ISBN 9780387310732.
29. Rubinstein, Y.D.; Hastie, T. Discriminative vs Informative Learning. In Proceedings of the Proceedings of the The Third International Conference on Knowledge Discovery and Data Mining; 1997; pp. 49–59.
30. Jain, A.K.; Duin, R.P.W.; Mao, J. Statistical pattern recognition: A review. *IEEE Trans. Pattern Anal. Mach. Intell.* **2000**, *22*, 4–37.
31. Raudys, S.; Duin, R.P.W. *Expected classification error of the Fisher linear classifier with pseudo-inverse covariance matrix*; 1998; Vol. 19;.
32. Burges, C.J.C. A Tutorial on Support Vector Machines for Pattern Recognition. *Data Min. Knowl. Discov.* **1998**, *2*, 121–167.
33. Müller, K.R.; Mika, S.; Rätsch, G.; Tsuda, K.; Schölkopf, B. An introduction to kernel-based learning algorithms. *IEEE Trans. Neural Networks* **2001**, *12*, 181–201.
34. Atiya, A.F. Estimating the posterior probabilities using the K-nearest neighbor rule. *Neural Comput.* **2005**, *17*, 731–740.
35. Bilucaglia, M.; Pederzoli, L.; Giroldini, W.; Prati, E.; Tressoldi, P. EEG correlation at a distance: A re-analysis of two studies using a machine learning approach. *F1000Research* **2019**, *8*.
36. Correia, J.M.; Jansma, B.; Hausfeld, L.; Kikkert, S.; Bonte, M. EEG decoding of spoken words in bilingual listeners: From words to language invariant semantic-conceptual representations. *Front. Psychol.* **2015**, *6*.
37. Duma, G.M.; Mento, G.; Semenzato, L.; Tressoldi, P. EEG anticipation of random high and low arousal faces and sounds. *F1000Research* **2018**, *8*, 1–15.
38. Mo, C.; Lu, J.; Wu, B.; Jia, J.; Luo, H.; Fang, F. Competing rhythmic neural representations of orientations during concurrent attention to multiple orientation features. *Nat. Commun.* **2019**, *10*.
39. Roberts, T.; Cant, J.S.; Nestor, A. Elucidating the Neural Representation and the Processing Dynamics of Face Ensembles. *J. Neurosci.* **2019**, *39*, 7737–7747.
40. Tang, J.; Alelyani, S.; Liu, H. Feature selection for classification: A review. In *Data Classification: Algorithms and Applications*; Aggarwal, C.C., Ed.; 2014; pp. 37–64 ISBN 9781466586758.
41. Miao, J.; Niu, L. A Survey on Feature Selection. *Procedia Comput. Sci.* **2016**, *91*, 919–926.
42. Müller, K.; Krauledat, M.; Dornhege, G.; Curio, G.; Blankertz, B. Machine learning techniques for brain-

computer interfaces. *Biomed. Tech. (Biomed. Tech.)* **2004**, *49*, 11–22.

43. Tottenham, N.; Tanaka, J.W.; Leon, A.C.; McCarry, T.; Nurse, M.; Hare, T.A.; Marcus, D.J.; Westerlund, A.; Casey, B.J.; Nelson, C. The NimStim set of facial expressions: Judgments from untrained research participants. *Psychiatry Res.* **2009**, *168*, 242–249.
44. Stevenson, R.A.; James, T.W. Affective auditory stimuli: characterization of the International Affective Digitized Sounds (IADS) by discrete emotional categories. *Behav. Res. Methods* **2008**, *40*, 315–21.
45. Stone, J. V. Independent component analysis: an introduction. *Trends Cogn. Sci.* **2002**, *6*.
46. Allen, J.J.B.; Coan, J.A.; Nazarian, M. Issues and assumptions on the road from raw signals to metrics of frontal EEG asymmetry in emotion. *Biol. Psychol.* **2004**, *67*, 183–218.
47. Babiloni, C.; Stella, G.; Buffo, P.; Vecchio, F.; Onorati, P.; Muratori, C.; Miano, S.; Gheller, F.; Antonaci, L.; Albertini, G.; et al. Cortical sources of resting state EEG rhythms are abnormal in dyslexic children. *Clin. Neurophysiol.* **2012**, *123*, 2384–2391.
48. Mert, A.; Akan, A. Emotion recognition from EEG signals by using multivariate empirical mode decomposition. *Pattern Anal. Appl.* **2018**, *21*, 81–89.
49. Clarke, A.R.; Barry, R.J.; Karamacoska, D.; Johnstone, S.J. The EEG Theta/Beta Ratio: A marker of Arousal or Cognitive Processing Capacity? *Appl. Psychophysiol. Biofeedback* **2019**, 1–7.
50. Blankertz, B.; Lemm, S.; Treder, M.; Haufe, S.; Müller, K.R. Single-trial analysis and classification of ERP components - A tutorial. *Neuroimage* **2011**, *56*, 814–825.