Applied Soft Computing 24 (2014) 212–221

Contents lists available at ScienceDirect

Applied Soft Computing

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

Data stream synchronization for defining meaningful fMRI classification problems

Marcin Budka *

Bournemouth University, Data Science Institute, Faculty of Science and Technology, Poole House, Talbot Campus, Fern Barrow, BH12 5BB Poole, UK

ARTICLE INFO

Article history:
Received 21 March 2013
Received in revised form 22 January 2014
Accepted 7 July 2014
Available online 15 July 2014

Keywords:
Pattern recognition
Machine learning
Classification
fMRI
Data stream synchronization
Smart filtering

ABSTRACT

Application of machine learning techniques to the functional Magnetic Resonance Imaging (fMRI) data is recently an active field of research. There is however one area which does not receive due attention in the literature – preparation of the fMRI data for subsequent modelling. In this study we focus on the issue of synchronization of the stream of fMRI snapshots with the mental states of the subject, which is a form of smart filtering of the input data, performed prior to building a predictive model. We demonstrate, investigate and thoroughly discuss the negative effects of lack of alignment between the two streams and propose an original data-driven approach to efficiently address this problem. Our solution involves casting the issue as a constrained optimization problem in combination with an alternative classification accuracy assessment scheme, applicable to both batch and on-line scenarios and able to capture information distributed across a number of input samples lifting the common simplifying i.i.d. assumption. The proposed method is tested using real fMRI data and experimentally compared to the state-of-the-art ensemble models reported in the literature, outperforming them by a wide margin.

Introduction

In the last years there has been a great deal of research on applying machine learning techniques and tools to processing of the functional Magnetic Resonance Imaging (fMRI) outputs [15,22,27]. The prospect of determining how mental states are mapped onto distributed patterns of neural activity is very attractive. The significance of this problem stems from countless potential applications in the area of neurology, Human–Machine/Brain–Computer Interfacing (HMI/BCI), or facilities for disabled and elderly people.

The research has been mostly organized around the following three key areas [19]: (1) application of classification methods to fMRI data (e.g. [1,8–11,23–26]), including combinations of classifiers also known as ensemble models (e.g. [17,28,29]), (2) dimensionality reduction techniques (e.g. [2,3,10,21,23,30]) and (3) spatio-temporal filtering (e.g. [18]). This research has been additionally stimulated and facilitated by the fMRI equipment becoming increasingly more accessible and affordable, the exponential increase in the available computational resources and unparalleled advances in the field of machine learning.

Since application of intelligent computational techniques to fMRI data is a relatively well developed area, in this study we focus on a more fundamental issue of the fMRI data preparation process and defining classification problems to be solved. According to data mining practitioners [20], data preparation[1] can take up to 80% of the modelling efforts and is crucial for development of well-performing models. In the current research however this issue is often overlooked or at best addressed heuristically. A central problem in our view is the alignment of the fMRI data with the actual mental states (i.e. data labelling), so that it is possible to develop a predictive model which not only demonstrates a certain level of accuracy but also solve a *real* classification problem. In a typical fMRI experiment a subject is instructed to enter a number of mental states in sequence (e.g. 'think of something funny', 'think of something sad') and the responses are captured with a certain, fixed sampling rate, resulting in a sequence of brain activity observations (snapshots). It is not guaranteed however that the subject will indeed enter the required mental state or how fast it will happen. Hence there is an inherent uncertainty to what extent a recorded brain snapshot corresponds to the actual desired mental state. In modelling of the Blood Oxygen Level Dependent (BOLD)

* Tel.: +44 793 52 33 571.
E-mail addresses: mbudka@bournemouth.ac.uk, mbudka@gmail.com

[1] In this context 'data preparation' is a term encompassing integration of data from multiple sources, sampling, labelling and standard preprocessing (e.g. data transformation, editing/imputation, attribute selection).

http://dx.doi.org/10.1016/j.asoc.2014.07.011

signal, this effect is additionally magnified by the haemodynamic response delay [13] and relatively low resolution of the images [27]. Lack of alignment of the fMRI data with mental states can contribute to the difficulty in translating between functional brain states of different subjects, so that a predictive model trained on data collected from one subject, would generalize well to others. The alignment issue is however often ignored and all snapshots taken while a certain stimulus is active are routinely averaged [27] (in some cases trimming 1–2 initial snapshots), or labelled with this particular stimulus [17]. While the former approach considerably reduces the size of the dataset, which is always small to begin with in relation to its dimensionality, it can also result in loss of valuable information and distorting it with noisy patterns. The latter, naïve labelling approach may however lead to defining a classification problem which is not meaningful (i.e. training data labels do not correspond to the mental states), rendering the subsequent analysis and modelling efforts futile or suboptimal at best. As an example, consider two brain snapshots taken while presenting two different stimuli A and B, which are more similar to each other than two snapshots taken when presenting stimulus C alone. As demonstrated later in this study, it is not an uncommon situation.

Automatic intelligent labelling of fMRI data is by no means a trivial task. Modelling of a fixed haemodynamic response function in the MRI literature [13,15] can be perceived as one attempt to address this issue, yet to the best of our knowledge no pure data-driven approach based on machine learning techniques exists. Hence in this paper we propose such an approach by viewing the problem of intelligent labelling of fMRI outputs as synchronization of the fMRI data stream and the label stream. We validate the proposed method on real, publicly available fMRI data and investigate its performance while using various measures of fit between the two data streams.

The main contribution of this paper is an original method for assigning labels to a stream of fMRI data, which results from challenging two popular approaches to fMRI data stream labelling for subsequent predictive model training. The proposed method is purely data-driven and lifts some of the restrictions of a typical signal processing based approach with a fixed haemodynamic response function. We also propose an alternative classification accuracy assessment scheme designed to make the results obtained with various fMRI data labellings more comparable.

The remainder of this paper is organized as follows. In Section "Problem setting" we further motivate and formally define the problem being addressed. We also give details of the data used in the experiments and perform its basic analysis. In Section "Baseline approach" a baseline approach taken from the literature is discussed and evaluated, with an in-depth analysis of its performance and identification of problematic areas. Section "Alternative accuracy assessment scheme and on-line predictions" presents an alternative model assessment scheme designed to make the results obtained with various fMRI data labellings more comparable and to take advantage of distributed representation of information in the on-line (i.e. prediction) mode. In Section "Synchronization criteria and optimization scheme" we present three measures of fit between the fMRI and label stream as well as an appropriate optimization algorithm for stream synchronization. The experimental results can be found in Section "Experimental results", while the conclusions have been given in Section "Conclusions and future research directions".

Problem setting

In this paper we use the data collected by Haxby within his seminal study 'Distributed and Overlapping Representations of Faces and Objects in Ventral Temporal Cortex' [14]. Haxby's experiment involved presenting 8 different visual stimuli to a single subject in 10 sessions. A snapshot of the brain has been taken every $TR = 2.5$ s, where TR denotes a discrete time point, and each session consisted of a single presentation of every stimulus in a random order for 9 *TRs*, followed by a 5-TR rest period. In the reminder of this paper we refer to the brain snapshots also as *TRs* or samples, while the stimulus to which a given sample refers to is called the class of the sample. The data we have used has been provided with the Princeton Multi-Voxel Pattern Analysis (MVPA) Toolbox[2] and the only preprocessing operation we have performed prior to our analysis was session-wise normalization (z-scoring).

While some of the issues resulting from labelling all samples taken while a certain stimulus is presented with this particular stimulus, and then using the obtained dataset to build and validate a classifier are well known (e.g. strong autocorrelation of samples violating the i.i.d. assumption on which most standard machine learning algorithms rely), there is another problem. In order to demonstrate it, we have calculated the within-class and between-class similarity of all collected samples (including the *rest* periods[3]). The results have been presented in Figs. 1 and 2[4] (the definitions of these two distances are given later in Section "Synchronization criteria and optimization scheme").

In Fig. 1(a) the median distance between samples coming from all pairwise combinations of the 9 classes (including the *rest* periods) has been depicted. As it can be seen there are cases, where the average within-class distance is much larger than the average between-class distance. For example, the median distance within the *face* class ($\approx$295) is higher than the median distance between *bottle* and *shoe* ($\approx$263) or *shoe* and *scramble* ($\approx$265). Although not as clearly visible in the case of the Cosine distance in Fig. 2(a), the median distance within the *scramble* class ($\approx$0.988) is still higher than the average distance between classes *house* and *shoe* ($\approx$0.982).

The above issue is even more pronounced in Figs. 1(b) and 2(b), where the minimal between-class distance has been plotted off the diagonal, while the diagonal entries represent the maximal within-class distance. As a result, by using the naïve labelling scheme one effectively expects the predictive model to correctly discriminate between items from different classes, which are more similar to each other than the items belonging to the same class – a rather risky and counterintuitive endeavour.

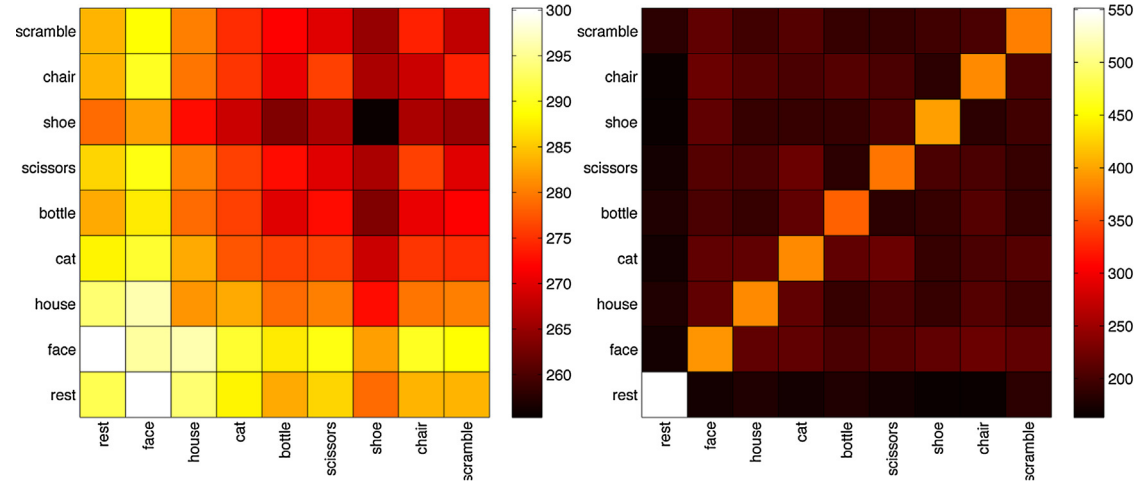
For the sake of completeness, we have also performed the same calculations for the case, when all snapshots taken during a single presentation of a stimulus are averaged. As it can be seen in Figs. 3 and 4, the situation did not improve much – one can still observe cases, in which samples belonging to a single class are less similar to each other than samples belonging to two different classes.

As mentioned before we propose to look at the fMRI outputs as a stream of data, which needs to be synchronized with a stream of labels. An example has been depicted in Fig. 5, where the arrows at the top represent the labels assigned to selected *TRs* within each stimulus presentation. Note, that unlike a fixed haemodynamic response lag, in the proposed setting the lag can be different for every presentation of each stimulus. This is an important feature of our approach due to the well-known variability of the haemodynamic response across subjects, sessions in a single subject and stimuli. Also, we allow multiple *TRs* in each presentation to be

[2] http://code.google.com/p/princeton-mvpa-toolbox/.

[3] The *rest* periods have been included here to demonstrate the high variability within a single class. In the classification experiments the *rest* periods have been discarded.

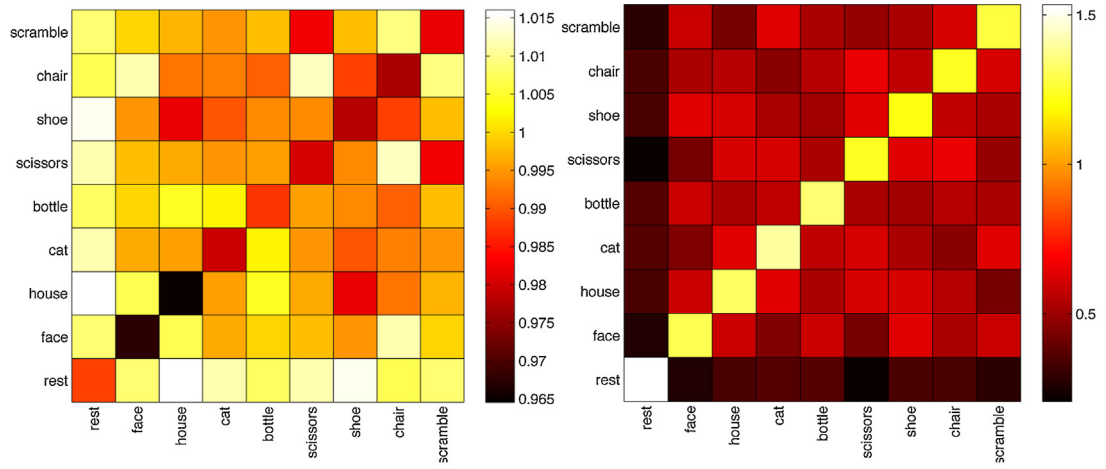
[4] All 43,193 voxels available in the dataset have been used for producing these figures. In order to alleviate the issue of potential concentration of the Euclidean distance (see [7]), concentration-resistant Cosine distance has also been used to confirm the findings.



(a) Median between-class and within-class distance

(b) Minimal between-class and maximal within-class distance

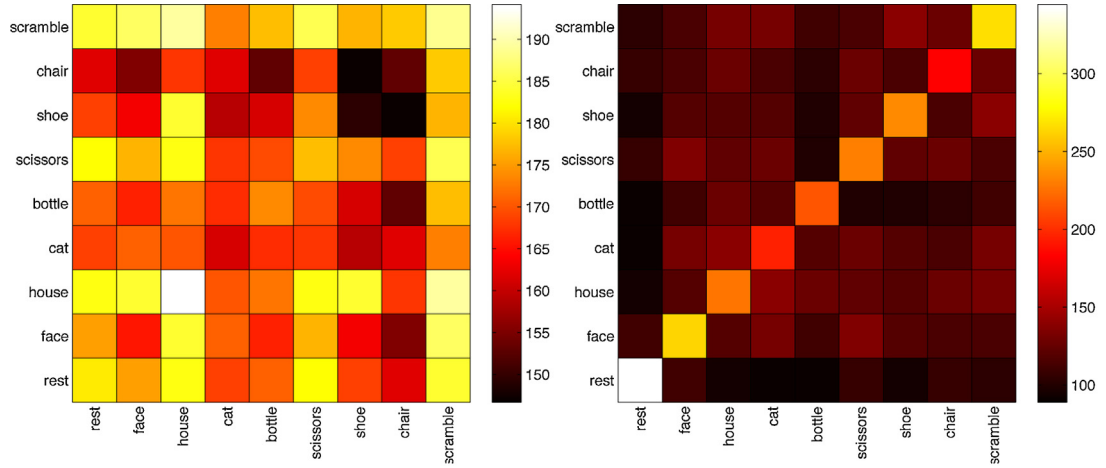
Fig. 1. Euclidean distance between classes representing visual stimuli, where each *TR* within a single presentation has been labelled with the presented stimulus.



(a) Median between-class and within-class distance

(b) Minimal between-class and maximal within-class distance

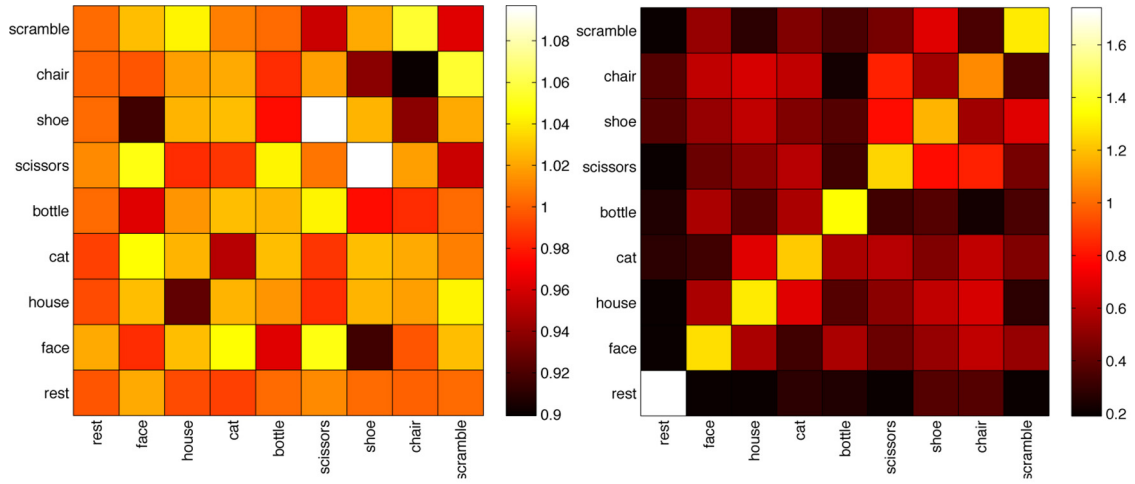
Fig. 2. Cosine distance between classes representing visual stimuli, where each *TR* within a single presentation has been labelled with the presented stimulus.



(a) Median between-class and within-class distance

(b) Minimal between-class and maximal within-class distance

Fig. 3. Euclidean distance between classes representing visual stimuli, where all *TR* within a single presentation have been averaged.



(a) Median between-class and within-class distance

(b) Minimal between-class and maximal within-class distance

Fig. 4. Cosine distance between classes representing visual stimuli, where all *TR* within a single presentation have been averaged.

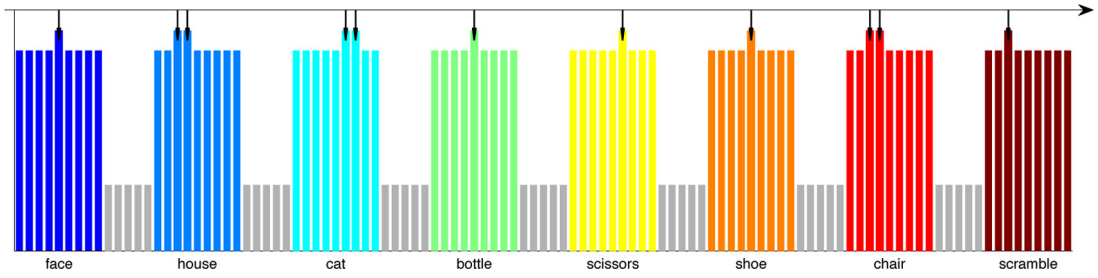


Fig. 5. fMRI and label streams.

selected and labelled, to account for the effect of sampling resolution artefacts (e.g. when the blood oxygenation level peaks between two consecutive samples or the peak spreads to a flat plateau).

Formally, the stream synchronization approach can be defined as a constrained optimization problem, where one is seeking such assignment of labels to the fMRI data stream, which minimizes a chosen criterion $\mathcal{J}$.

Denoting by $\boldsymbol{x}_i$ the ith vector of voxel activations (sample), by L_i the stimulus shown while $\boldsymbol{x}_i$ was recorded (class), the input dataset can be defined as a sequence of sample/label pairs: $D = ((\boldsymbol{x}_1, L_1), (\boldsymbol{x}_2, L_2), \ldots, (\boldsymbol{x}_N, L_N))$, where N is the number of recorded snapshots. Let $I = (i_1, i_2, \ldots, i_N)$ be a binary selector vector and $k \in \{1, 2, \ldots, N\}$. We define a filter function $\mathcal{F}$ given by $D_F = \mathcal{F}(D, I)$, where $D_F = ((\boldsymbol{x}_k, L_k) : i_k = 1)$. The optimization problem then becomes:

$$\begin{aligned} &\underset{I}{\operatorname{argmin}} \quad \mathcal{J}(\mathcal{F}(D, I)) \\ &\text{s.t.} \qquad \forall i \in \left\{1, 2, \ldots, \frac{N}{P}\right\} : \sum_{l=(i-1)\times P+1}^{i\times P} i_l \geq 1 \end{aligned} \tag{1}$$

where P is the length of stimulus presentation in *TRs* (in our case $P = 9$) and the set of constraints ensures that at least one sample from each stimulus presentation is included in D_F.

Although the approach described above can be perceived as a variant of temporal feature selection (if all *TRs* from a single stimulus presentations were treated as a single sample), it has an important advantage – compatibility with standard machine learning algorithms. Most machine learning algorithms have been designed to handle input vectors of a fixed size, where the meaning of each element of these vectors does not change over time. This has an important consequence for what is traditionally understood as temporal feature selection: the number of features selected must be the same for all samples (e.g. one always selects exactly 3 *TRs*). As this can be suboptimal, the method proposed in this paper does not have such restrictions – the number of selected *TRs* can be anything between 1 and P.

Baseline approach

As a starting point we have attempted to reproduce the best results reported in [17], i.e. 73.2% 10-fold cross-validation[5] accuracy of a Random Forest ensemble with 1000 trees. Following [17], we have first selected a subset of voxels by cross-training[6] 10 Support Vector Machines (SVMs) with linear kernels and then extracting top 200 contribution weights of voxels in terms of their absolute value, from each model. The intersection of the 10 sets obtained in this way resulted in 92 voxels common for all sessions (Fig. 6), which is a slightly different result when compared to [17], where the authors have reported the intersection to contain 93 voxels. We have attributed this to slight differences in the SVM

[5] Cross-validation (CV) is a standard statistical technique for estimation of model generalization ability, i.e. assessing how the model will perform on new, previously unseen data [16]. In k-fold cross-validation the dataset is randomly divided into k approximately equal subsets. Each subset (called 'fold') is then in turn put aside as validation data, a model is built using the remaining $k-1$ folds and tested on the validation fold. The error estimate is then calculated as a mean of all validation errors. In both [17] and this study each stimulus presentation session corresponds to a single cross-validation fold.

[6] In k-fold cross-training each of the k base models is trained on the union of $k-1$ folds, every time leaving a different fold aside.

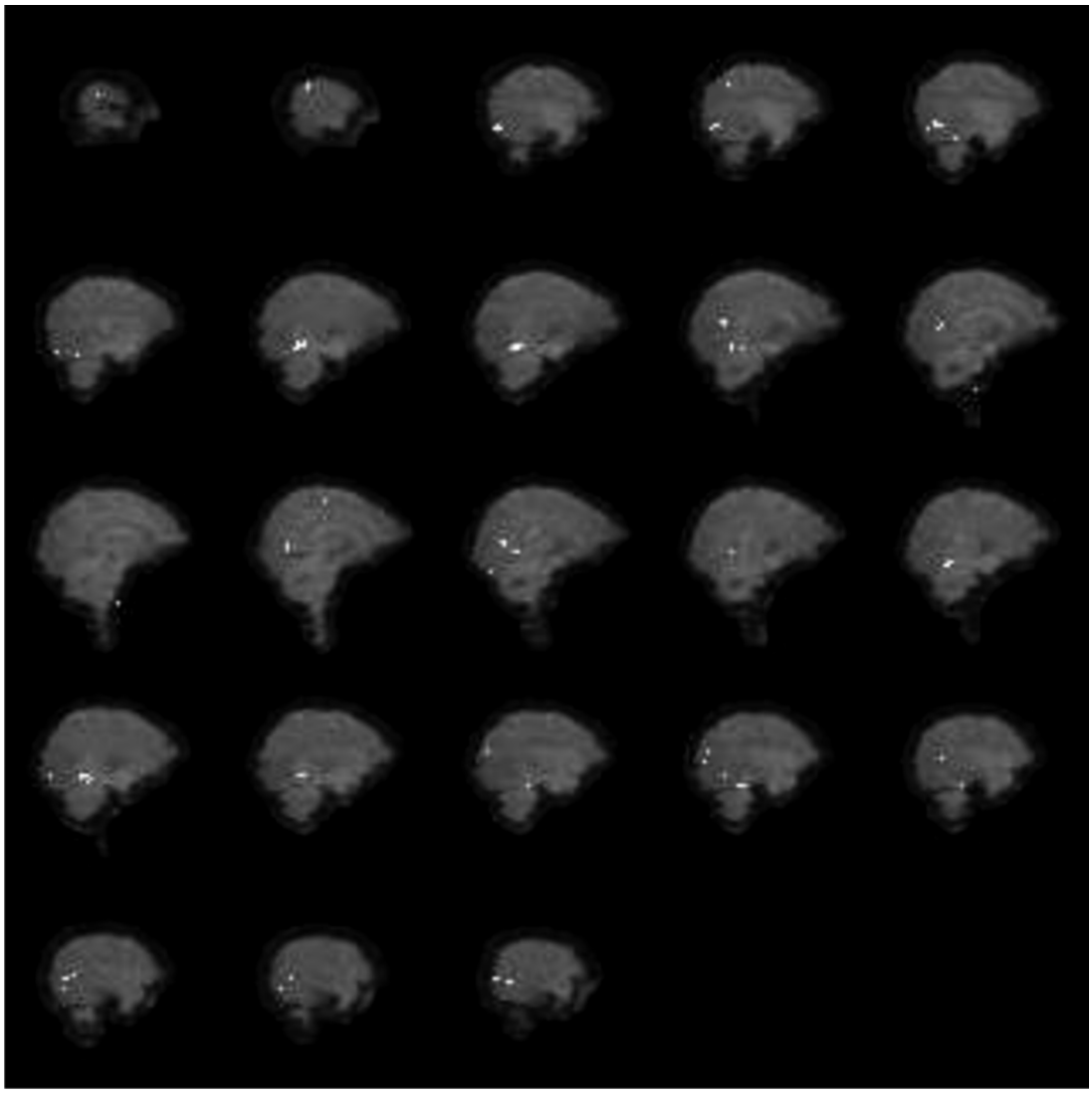


Fig. 6. Voxels common for all 10 sessions.

Table 1
Classifier list.

Name	Description
ldc	Linear Discriminant Classifier[a]
qdc	Quadratic Discriminant Classifier[b]
knnc	*K*-Nearest Neighbour Classifier[c]
svc	Support Vector Classifier with linear kernels[d]

[a,b] The regularization parameters of the classifiers have been optimized automatically using a built-in routine of the PRTools toolbox (internal 5-fold cross-validation).
[c] The parameter *K* (the number of nearest neighbours) has been optimized automatically using a built-in routine of the PRTools toolbox (internal leave-one-out cross-validation).
[d] The default value of the regularization parameter has been used.

implementations from various authors. In our experiments a Support Vector Classifier (*svc*) from the PRTools Pattern Recognition Toolbox version 4.2.1 for MATLAB [12] has been used.

Rather than building complex ensemble models, we have first opted for a set of basic classifiers included in the PRTools toolbox – their list has been given in Table 1, while their classification accuracy assessed using the same approach as in [17] has been reported in Table 2.

As it can be seen, a simple linear classifier (*ldc*) was able to achieve an average accuracy of 74.2%, not only vastly outperforming other tested classifiers, but also outperforming the Random Forest ensemble from [17] at the fraction of computations, yet still leaving room for improvement. Hence all the remaining experiments reported in this paper are performed using *ldc* as a base model and focus on the influence of label stream synchronization on the classification accuracy rather than on optimization and tuning of the classifier itself.

Fig. 7 presents the *TRs* for which *ldc* produced incorrect predictions (marked with 'x'), broken down into session/stimulus pairs. First thing to notice is that some classes seem not to pose problems to the classifier (*house*, *chair* or *scramble*) while other appear to be rather difficult to handle (*bottle*, *scissors*, *cat* or *shoe*).

The confusion matrix given in Table 3 provides some insight into this situation:

- *bottle* is often confused with both *scissors* (12 *TRs*) and *shoe* (13 *TRs*) as well as *chair* (8 *TRs*),
- *scissors* are mostly confused with *bottle* (11 *TRs*) and *scramble* (8 *TRs*),
- *shoe* is mostly confused with *bottle* (16 *TRs*).

From the above it is apparent that the *bottle–scissors–shoe* trio is causing most of the trouble here as the combination of selected

Table 2
Classification accuracy – 200 pre-selected voxels.

Fold→	1	2	3	4	5	6	7	8	9	10	Mean
ldc	80.6	65.3	86.1	80.6	66.7	81.9	70.8	73.6	68.1	68.1	**74.2**
qdc	61.1	65.3	73.6	70.8	62.5	70.8	65.3	37.5	59.7	41.7	60.8
knnc	56.9	40.3	61.1	73.6	54.2	54.2	51.4	27.8	55.6	30.6	50.6
svc	69.4	61.1	72.2	80.6	52.8	62.5	55.6	54.2	65.3	61.1	63.5

The best result is given in bold.

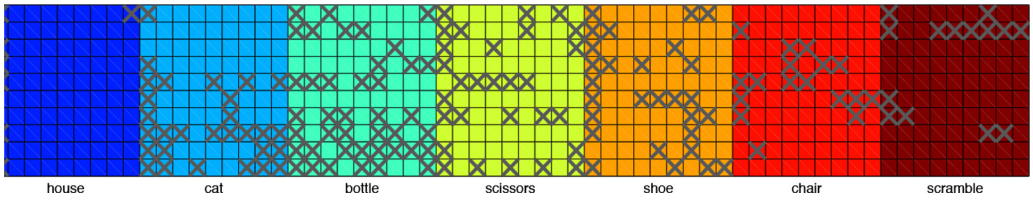


Fig. 7. Errors of *ldc*.

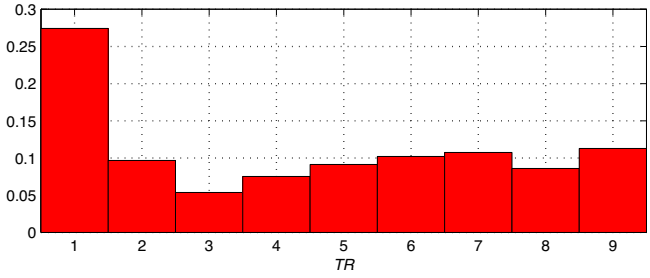


Fig. 8. *ldc* errors vs. *TR*.

features and classifier seem not to have enough discriminative power to tell them apart.

The histogram of classification errors vs. *TR* given in Fig. 8 provides further interesting insight: the first *TR* of each stimulus presentation is by far the most difficult to classify as it accounts for over 25% of all errors. This can be caused by natural variability in the onset of the haemodynamic response function or the initial dip in the BOLD response reported in some studies (for example [31] and references therein), resulting in a rather noisy pattern at the output of the fMRI device. On the other hand, the third and fourth *TRs* (between 5 and 10 s) appear to be the easiest to classify on average, which is also more or less consistent with the effect of the haemodynamic lag causing the blood oxygenation level to peak around 5 s after stimulus presentation [13].

At this stage it would thus be instructive to develop the following additional classification models and assess their performance:

- A model which uses all *TRs* except the first, most problematic *TR* of each stimulus presentation,
- A model which uses only the third, least problematic *TR* of each stimulus presentation,

Table 3
ldc confusion matrix.

True class	Predicted class							
	Face	House	Cat	Bottle	Scissors	Shoe	Chair	Scramble
Face	**69**	0	12	4	1	2	0	2
House	0	**83**	1	2	0	0	1	3
Cat	10	0	**61**	8	3	2	1	5
Bottle	1	1	2	**49**	12	13	8	4
Scissors	1	0	0	11	**61**	6	3	8
Shoe	1	0	0	16	2	**62**	5	4
Chair	0	0	0	13	1	2	**73**	1
Scramble	0	0	1	8	1	1	3	**76**

The diagonal entries have been given in bold to make it easier to visually inspect the confusion matrix for errors.

- A model which uses both the third and fourth, i.e. two least problematic *TRs* of each stimulus presentation.

The problem is however, that by dropping data corresponding to all *TRs* but the third, we would severely affect the validation mechanism. The reason for this is that rather than having 9 validation *TRs* for every stimulus in every session, we would end up with a single *TR* per stimulus. Moreover this would also make the results difficult to interpret and to compare between the experiments with different numbers of validation *TRs*. Hence in the next section we propose an alternative model assessment scheme, which levels the ground for all models, regardless of the actual number of *TRs* used for training.

Alternative accuracy assessment scheme and on-line predictions

In order to make the experiments comparable between models trained using different number of *TRs*, we have devised the following alternative accuracy assessment scheme, embedded into standard cross-validation.

For every fold, a model is built using only the selected *TRs* from the 9 training folds, but for validation all *TRs* from the remaining fold are always used. We are however not interested in assessing prediction for every single *TR* of the validation data but rather for a group of *TRs*, which correspond to presentation of a single stimulus. Hence for every such group of 9 *TRs* (batch) we produce a single classification decision by summing the soft outputs[7] of the classifier and selecting the class for which this sum is maximized. This allows to capture information distributed across neighbouring *TRs* rather than using only a single *TR* for casting a prediction. It also ensures robustness of our approach, as the influence of noisy samples is minimized through the way in which the classifier outputs reflect uncertainty (i.e. there is no dominating class). The procedure has been depicted in Fig. 9.

The classification accuracy obtained using the above scheme has been reported in Table 4, where the subscripts in the leftmost column denote the *TRs* of each presentation used for training. As it can be seen, the best results – 85% accuracy – have been obtained when using only *TRs* # 3 and # 4 or not using *TR* # 1. Hence 85% accuracy becomes our new baseline.

The proposed accuracy assessment scheme can be easily extended to support a true on-line scenario, in which consecutive samples arrive one by one and are classified in real-time. The results

[7] *ldc* produces so called 'soft' or 'fuzzy' outputs, which denote the degree of membership of a given sample to each class.

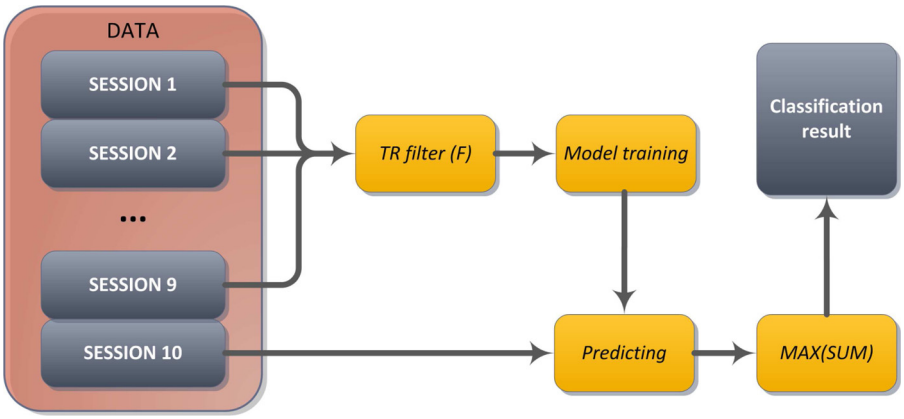


Fig. 9. Alternative accuracy assessment scheme.

of such experiment have been included in Section "Experimental results".

Synchronization criteria and optimization scheme

We propose to use the following optimization criteria in Eq. (1):

1. The ratio of the average intra-class distance to the average inter-class distance. This is a measure inspired by clustering algorithms [4], designed to encourage formation of groups of samples, which are similar to each other while dissimilar to the samples in other groups. The criterion is given by the following formula:

$$\mathcal{J}_{dist} = \frac{C \times \sum_{c=1}^{C} \sum_{i=1}^{N_c} \sum_{j=1}^{N_c} d(\boldsymbol{x}_{ci}, \boldsymbol{x}_{cj})}{\sum_{c_1=1}^{C} \sum_{c_2=1}^{C} \sum_{i=1}^{N_{c_1}} \sum_{j=1}^{N_{c_2}} d(\boldsymbol{x}_{c_1 i}, \boldsymbol{x}_{c_2 j})} \tag{2}$$

   where C denotes the number of classes (stimuli), N_c is the number of selected *TRs* of the cth stimulus presentation, x_{ci} is the ith selected sample within cth presentation and $d(\cdot)$ is some distance measure, in our case:

   (a) the Euclidean distance: $d_E(\boldsymbol{x}_i, \boldsymbol{x}_j) = \sqrt{(\boldsymbol{x}_i - \boldsymbol{x}_j)^T (\boldsymbol{x}_i - \boldsymbol{x}_j)}$

   (b) the Cosine distance: $d_C(\boldsymbol{x}_i, \boldsymbol{x}_j) = 1 - \frac{\boldsymbol{x}_i^T \boldsymbol{x}_j}{\sqrt{\boldsymbol{x}_i^T \boldsymbol{x}_i} \times \sqrt{\boldsymbol{x}_j^T \boldsymbol{x}_j}}$

2. The internal cross-validation error calculated within *ldc* during the regularization parameter tuning $\mathcal{J}_{cv}$.

In order to solve the problem defined by Eq. (1) we have devised an iterative greedy optimization scheme given by Algorithm 1. In our experience greedy approaches work surprisingly well (see [5,6] for example), often outperforming stochastic methods like genetic algorithms or simulated annealing, when a strict time limit is imposed on the optimization process. Also, due to their deterministic nature, all experiments are easily reproducible and there is no need to account for random factors when comparing the results of multiple experiments.

Algorithm 1.

Optimize $\mathcal{J}$.

Initialization:
  $D \leftarrow ((\boldsymbol{x}_1, L_1), (\boldsymbol{x}_2, L_2), \ldots, (\boldsymbol{x}_N, L_N))$
  $I \leftarrow [001000000\ 001000000 \ldots 001000000]$
  $bestScore \leftarrow \mathcal{J}(\mathcal{F}(D, I))$
  $improvement \leftarrow false$
Optimization:
  while $improvement = false$ **do**
    for all $sp \leftarrow$ stimuli presentations **do**
      for all $iTR \leftarrow (I(sp) = 0)$ **do**
        $I_{temp} \leftarrow I$
        $I_{temp}(sp) \leftarrow false$
        $I_{temp}(sp(iTR)) \leftarrow true$
        $score \leftarrow \mathcal{J}(\mathcal{F}(D, I_{temp}))$
        if $score < bestScore$ **then**
          $I \leftarrow It$
          $bestScore \leftarrow score$
          $improvement \leftarrow true$
        end if
      end for
    end for
  end while

In Algorithm 1 we start by initializing all the relevant variables (note, that we are using symbols and terminology introduced in Section "Problem setting"). This includes the binary selector vector I, which initially selects the 3rd *TR* of each stimulus presentation, according to our earlier argument in Section "Baseline approach". Then, in the main loop we iteratively test various label assignments, accepting a new one only if it is better than all assignments tested before. The optimization algorithm is run 10 times following the cross-validation scheme and hence results in 10 versions of the selector vector I, with a single active *TR* per stimulus presentations. We then try to improve the assignment by re-running a slightly

Table 4
Classification accuracy of *ldc* – 200 pre-selected voxels and alternative performance assessment scheme.

Fold→	1	2	3	4	5	6	7	8	9	10	Mean
TR_{1-9}	62.5	62.5	87.5	100	75.0	87.5	75.0	75.0	62.5	62.5	75.0
TR_{2-9}	87.5	62.5	100	100	100	87.5	75.0	75.0	62.5	100	**85.0**
TR_3	87.5	62.5	87.5	100	75.0	75.0	87.5	87.5	62.5	75.0	80.0
TR_4	87.5	75.0	87.5	75.0	87.5	87.5	75.0	87.5	62.5	87.5	81.3
$TR_{3,4}$	75.0	87.5	87.5	100	87.5	87.5	75.0	100	62.5	87.5	**85.0**

The best result is given in bold.

Table 5
Classification accuracy of *ldc* – 200 pre-selected voxels, alternative assessment scheme and various stream synchronization criteria.

Fold→	1	2	3	4	5	6	7	8	9	10	Mean
$\mathcal{J}_{dist}/d_E$											
$TR_{(1)}$	100	87.5	87.5	87.5	87.5	87.5	100	87.5	75.0	62.5	**86.3**
$TR_{3,4+(1)}$	75.0	87.5	87.5	100	87.5	75.0	75.0	100.0	75.0	87.5	85.0
$TR_{3,4+(2)}$	87.5	87.5	87.5	100	75.0	87.5	75.0	100.0	62.5	87.5	85.0
$TR_{3,4+(3)}$	87.5	87.5	87.5	100	87.5	87.5	75.0	100.0	87.5	87.5	**88.8**
$\mathcal{J}_{dist}/d_C$											
$TR_{(1)}$	100	87.5	100.0	100	87.5	87.5	87.5	62.5	75.0	62.5	85.0
$TR_{3,4+(1)}$	75.0	87.5	87.5	100	87.5	87.5	87.5	87.5	75.0	87.5	**86.3**
$TR_{3,4+(2)}$	87.5	87.5	87.5	100	87.5	87.5	62.5	87.5	75.0	87.5	85.0
$TR_{3,4+(3)}$	87.5	75.0	87.5	100	100.0	87.5	87.5	100.0	62.5	87.5	**87.5**
$\mathcal{J}_{cv}$											
$TR_{(1)}$	87.5	62.5	87.5	100	62.5	75.0	100	62.5	62.5	75.0	77.5
$TR_{3,4+(1)}$	75.0	75.0	87.5	100	87.5	87.5	75.0	87.5	75.0	100	85.0
$TR_{3,4+(2)}$	75.0	75.0	87.5	100	87.5	87.5	75.0	87.5	75.0	100	85.0
$TR_{3,4+(3)}$	87.5	75.0	87.5	100	87.5	87.5	75.0	87.5	75.0	100	**86.3**

modified algorithm, this time initializing I with the result of the previous run, until no further improvement is observed.

Experimental results

The experiments have been performed in two tracks: starting with 200 pre-selected voxels, as described in Section "Baseline approach" (i.e. 10 sets of voxels selected within a 10-fold cross-validation scheme), and starting with all voxels (i.e. without the voxel pre-selection mechanism). Note that this applies to the stream synchronization criteria only, as in both cases the *ldc* model has been trained using the 200 pre-selected voxels due to computational tractability. For the same reason in the case of $\mathcal{J}_{cv}$ the experiments on all voxels have not been performed.

Tables 5 and 6 present the results for the two tracks of experiments. For each of the three synchronization criteria (e.g. $\mathcal{J}_{dist}/d_E$, $\mathcal{J}_{dist}/d_C$ and $\mathcal{J}_{cv}$) we have first allowed for a single pass of Algorithm 1, in order to assign the class labels to a single *TR* within each stimulus presentation only. In Tables 5 and 6 this has been denoted by $TR_{(1)}$. The result was a very high 87.5% classification accuracy when using $\mathcal{J}_{dist}/d_E$ on all voxels, which is already above the baseline defined in Section "Alternative accuracy assessment scheme and on-line predictions" (for convenience, if the average accuracy in Tables 5 and 6 exceeded the baseline accuracy of 85%, it has been typed in bold).

Since as a greedy approach, the proposed optimization scheme is susceptible to getting caught in local minima, in the next experiment we have used a known good solution as a starting point – the algorithm was initialized by pre-selecting *TRs* # 3 and # 4 of each presentation according to the results reported in Table 4. We have then run the optimization 3 more times in sequence, every time feeding the result of the previous run as an input. This way we have allowed the algorithm to select up to 3, 4 and 5 *TRs* of each stimulus presentation ($TR_{3,4+(1)}$, $TR_{3,4+(2)}$ and $TR_{3,4+(3)}$ in Tables 5 and 6). The best result in our experiments – 88.8% accuracy – has been obtained for the combination of $\mathcal{J}_{dist}/d_E$ and $TR_{3,4+(3)}$. Although we believe the results could be further improved, for example by employing ensemble models as discussed in [17], we consider that the point has been proven and rather focus on examining the best model in more detail.

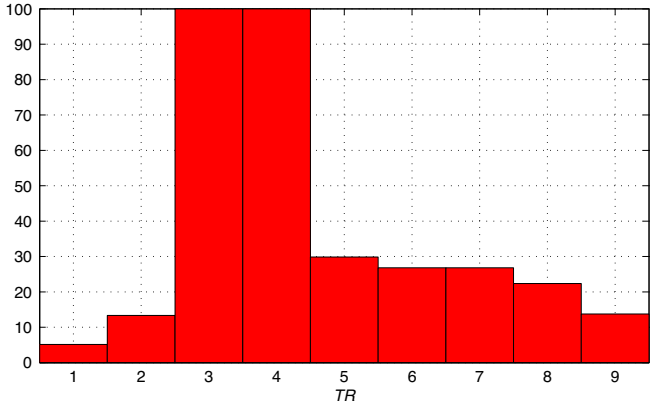


Fig. 10. *TRs* used by the best model.

Fig. 10 shows a histogram of the *TRs* used by the best model. The values on the $y-$axis denote the percentage of stimulus presentations in which a given *TR* was labelled. As it can be seen, *TRs* # 3 and # 4 are absolutely crucial in this case as they have always been used in all presentations, complemented by *TRs* # 5, # 6 and # 7 in ≈30% of cases but also by *TR* # 1 in less than 5% of the cases. Although according to our previous argument *TR* # 1 is by far the most problematic, as it can be seen, in some situations its inclusion is beneficial. This last observation emphasizes our claim that the

Table 6
Classification accuracy of *ldc* – all voxels, alternative assessment scheme and various stream synchronization criteria.

Fold→	1	2	3	4	5	6	7	8	9	10	Mean
$\mathcal{J}_{dist}/d_E$											
$TR_{(1)}$	100.0	87.5	87.5	87.5	100	87.5	100	87.5	75.0	62.5	**87.5**
$TR_{3,4+(1)}$	75.0	87.5	87.5	100	75.0	87.5	87.5	87.5	75.0	87.5	85.0
$TR_{3,4+(2)}$	75.0	87.5	100	100	100	87.5	75.0	87.5	62.5	87.5	**86.3**
$TR_{3,4+(3)}$	87.5	75.0	87.5	87.5	100	100	62.5	87.5	62.5	75.0	82.5
$\mathcal{J}_{dist}/d_C$											
$TR_{(1)}$	25.0	100	87.5	75.0	87.5	75.0	62.5	62.5	75.0	62.5	71.3
$TR_{3,4+(1)}$	75.0	100	87.5	100.0	75.0	87.5	87.5	75.0	75.0	100	**86.3**
$TR_{3,4+(2)}$	75.0	100	62.5	87.5	87.5	87.5	75.0	75.0	62.5	87.5	80.0
$TR_{3,4+(3)}$	75.0	87.5	87.5	87.5	100	87.5	75.0	75.0	62.5	87.5	82.5

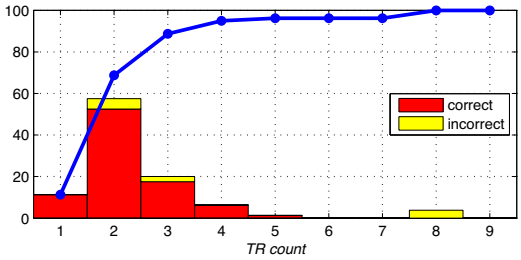


Fig. 11. Probability of stable classification decision vs. the number of *TRs* seen.

fMRI data should be labelled in a flexible, data-driven way rather than by imposing fixed requirements, for example of not including *TR* # 1 in the analysis at all (trimming).

An interesting thing to note from the presented results is that the data from some sessions seem to be much easier to classify than from others. In Table 5, session 4 is one such example, where the accuracy is seldom below 100%, while in case of session 9 the accuracy never exceeds 75%. Although beyond the scope of this study, it might be instructive to closely examine the setup of the fMRI experiment in [14] looking for an explanation of this phenomenon.

In the last experiment presented in this section we have investigated extending the accuracy assessment scheme proposed in Section "Alternative accuracy assessment scheme and on-line predictions" for on-line predictions of class labels of incoming samples. In this scenario we count how many consecutive *TRs* must be seen by the model for the classification decision to stabilize, i.e. not change when more samples from the same batch arrive. The results have been depicted in Fig. 11.

First thing to note is that almost 70% of classification decisions are made after examining just 2 incoming samples (*TRs*). Moreover, most of these decisions are correct and 3 incoming samples suffice to reach the final decision in about 90% of the cases. In just under 4% of cases the decisions require examination of as much as 8 samples, but this is where the classifier is never correct.

Conclusions and future research directions

Recent advances in the area of machine learning, together with the ever increasing computational power of affordable equipment allow to model almost any relationship underlying a given dataset, regardless of how spurious it is. Definition of *meaningful* learning problems hence arises as an important issue, relevant to any data-intensive discipline. In this paper we have explored and addressed this issue in the context of fMRI data modelling.

To this end, we have challenged two popular approaches to fMRI data stream labelling for subsequent predictive model training, demonstrating some of their weaknesses. The solution we have proposed involves casting the issue as a constrained optimization problem in combination with an alternative classification accuracy assessment scheme, applicable to both batch and on-line scenarios and able to capture information distributed across a number of input samples lifting the common simplifying i.i.d. assumption.

By employing the proposed fMRI data labelling approach and redefining the standard classification problem in terms of what we are really interested to predict, we have been able to take the initial 75% classification accuracy up to almost 89%. In the context of the dataset used in this study it translates to misclassification of only 9 out of 80 presented stimuli rather than 20 out of 80 without the stream synchronization algorithm. As noted in Section "Experimental results" this result could likely be further improved by employing more advanced modelling techniques like the ensemble models, which forms one of the future research directions.

The work can also be extended by redefining the optimization problem in order to make the objective function continuous and differentiable e.g. by relaxation. This would allow to use other, potentially more efficient optimization methods which rely on gradient information (e.g. quasi-Newton). Yet another direction we would like to pursue in the future is making the proposed approach fully incremental, to enable it to learn on the fly as new data arrives.

References

[1] K. Allen, F. Pereira, M. Botvinick, A.E. Goldberg, Distinguishing grammatical constructions with fMRI pattern analysis, Brain Lang. 123 (2012) 174–182.
[2] F. Bießmann, F. Meinecke, A. Gretton, A. Rauch, G. Rainer, N. Logothetis, K. Müller, Temporal kernel CCA and its application in multimodal neuronal data analysis, Mach. Learn. 79 (2010) 5–27.
[3] G. Blanchard, M. Kawanabe, M. Sugiyama, V. Spokoiny, K. Müller, In search of non-Gaussian components of a high-dimensional distribution, J. Mach. Learn. Res. 7 (2006) 247–282.
[4] M. Budka, Clustering as an example of optimizing arbitrarily chosen objective functions, in: N.T. Nguyen, B. Trawinski, R. Katarzyniak, G.S. Jo (Eds.), Advanced Methods for Computational Collective Intelligence, Studies in Computational Intelligence, vol. 457, Springer, Berlin/Heidelberg, 2013, pp. 177–186.
[5] M. Budka, B. Gabrys, Correntropy-based density-preserving data sampling as an alternative to standard cross-validation, in: The 2010 International Joint Conference on Neural Networks (IJCNN), 2010 a, pp. 1–8.
[6] M. Budka, B. Gabrys, Ridge regression ensemble for toxicity prediction, Proc. Comput. Sci. 1 (2010) 193–201.
[7] M. Budka, B. Gabrys, Electrostatic field framework for supervised and semi-supervised learning from incomplete data, Nat. Comput. 10 (2011) 921–945, http://dx.doi.org/10.1007/s11047-010-9182-4.
[8] C. Davatzikos, K. Ruparel, Y. Fan, D.G. Shen, M. Acharyya, J.W. Loughead, R.C. Gur, D.D. Langleben, Classifying spatial patterns of brain activity with machine learning methods: application to lie detection, NeuroImage 28 (3) (2005) 663–668, http://dx.doi.org/10.1016/j.neuroimage.2005.08.009.
[9] F. De Martino, F. Gentile, F. Esposito, M. Balsi, F. Di Salle, R. Goebel, E. Formisano, Classification of fMRI independent components using ic-fingerprints and support vector machine classifiers, Neuroimage 34 (2007) 177–194.
[10] F. De Martino, G. Valente, N. Staeren, J. Ashburner, R. Goebel, E. Formisano, Combining multivariate voxel selection and support vector machines for mapping and classification of fMRI spatial patterns, Neuroimage 43 (2008) 44.
[11] N. Dosenbach, B. Nardos, A. Cohen, D. Fair, J. Power, J. Church, S. Nelson, G. Wig, A. Vogel, C. Lessov-Schlaggar, et al., Prediction of individual brain maturity using fMRI, Science 329 (2010) 1358–1361.
[12] R. Duin, P. Juszczak, P. Paclik, E. Pekalska, D. de Ridder, D. Tax, S. Verzakov, PR-Tools 4.1, A MATLAB Toolbox for Pattern Recognition, 2007 http://prtools.org
[13] G. Glover, Deconvolution of impulse response in event-related BOLD fMRI, Neuroimage 9 (1999) 416–429.
[14] J. Haxby, M. Gobbini, M. Furey, A. Ishai, J. Schouten, P. Pietrini, Distributed and overlapping representations of faces and objects in ventral temporal cortex, Science 293 (2001) 2425–2430.
[15] J. Haynes, G. Rees, Decoding mental states from brain activity in humans, Nat. Rev. Neurosci. 7 (2006) 523–534.
[16] R. Kohavi, A study of cross-validation and bootstrap for accuracy estimation and model selection Proceedings of the 14th International Joint Conference on Artificial Intelligence, vol. 2, Morgan Kaufmann, San Francisco, 1995, pp. 1137–1145.
[17] L.I. Kuncheva, J.J. Rodríguez, Classifier ensembles for fMRI data analysis: an experiment, Magn. Reson. Imaging 28 (2010) 583–593.
[18] S. Lemm, B. Blankertz, G. Curio, K. Muller, Spatio-spectral filters for improving the classification of single trial EEG, IEEE Trans. Biomed. Eng. 52 (2005) 1541–1548.
[19] S. Lemm, B. Blankertz, T. Dickhaus, K. Müller, Introduction to machine learning for brain imaging, Neuroimage 56 (2011) 387–399.
[20] G. Linoff, M. Berry, Data Mining Techniques: For Marketing, Sales, and Customer Relationship Management, Wiley, Indianapolis, 2011.
[21] M. Mørup, L. Hansen, S. Arnfred, L. Lim, K. Madsen, Shift invariant multilinear decomposition of neuroimaging data, Neuroimage 42 (2008) 1439–1450.
[22] K. Norman, S. Polyn, G. Detre, J. Haxby, Beyond mind-reading: multi-voxel pattern analysis of fMRI data, Trends Cogn. Sci. 10 (2006) 424–430.
[23] A. O'Toole, F. Jiang, H. Abdi, N. Pénard, J. Dunlop, M. Parent, Theoretical, statistical, and practical perspectives on pattern-based classification approaches to the analysis of functional neuroimaging data, J. Cogn. Neurosci. 19 (2007) 1735–1752.
[24] F. Pereira, M. Botvinick, Classification of functional magnetic resonance imaging data using informative pattern features, in: Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, ACM, New York, 2011, pp. 940–946.
[25] F. Pereira, M. Botvinick, Information mapping with pattern classifiers: a comparative study, Neuroimage 56 (2011) 476–496.
[26] F. Pereira, M. Botvinick, A systematic approach to extracting semantic information from functional MRI data, in: Advances in Neural Information Processing Systems, vol. 25, 2012, pp. 2276–2284.

[27] F. Pereira, T. Mitchell, M. Botvinick, Machine learning classifiers and fMRI: a tutorial overview, Neuroimage 45 (2009) S199–S209.
[28] C. Plumpton, L. Kuncheva, D. Linden, S. Johnston, On-line fMRI data classification using linear and ensemble classifiers, in: Proceedings of the 2010 20th International Conference on Pattern Recognition, 2010, pp. 4312–4315.
[29] C.O. Plumpton, L.I. Kuncheva, N.N. Oosterhof, S.J. Johnston, Naive random subspace ensemble with linear classifiers for real-time classification of fMRI data, Pattern Recogn. 45 (2012) 2101–2108.
[30] J. Poppenk, K.A. Norman, Mechanisms supporting superior source memory for familiar items: a multi-voxel pattern analysis study, Neuropsychologia 50 (2012) 3015–3026.
[31] E. Yacoub, A. Shmuel, J. Pfeuffer, V. De Moortele, G. Adriany, K. Ugurbil, X. Hu, et al., Investigation of the initial dip in fMRI at 7 tesla, NMR Biomed. 14 (2001) 408–412.