

Decoding fMRI activity in the time domain improves classification performance

João Loula, Gaël Varoquaux, Bertrand Thirion

▶ **To cite this version:**

João Loula, Gaël Varoquaux, Bertrand Thirion. Decoding fMRI activity in the time domain improves classification performance. NeuroImage, 2017, 10.1016/j.neuroimage.2017.08.018 . hal-01576641

HAL Id: hal-01576641
https://inria.hal.science/hal-01576641

Submitted on 25 Aug 2017

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Decoding fMRI activity in the time domain improves classification performance

João Loula[a,1], Gaël Varoquaux[a], Bertrand Thirion[a]

[a] *Parietal Team - Inria / CEA - Paris Saclay University, France*
[b] *Department of Computer Science - École Polytechnique, France*

Abstract

Most current functional Magnetic Resonance Imaging (fMRI) decoding analyses rely on statistical summaries of the data resulting from a deconvolution approach: each stimulation event is associated with a brain response. This standard approach leads to simple learning procedures, yet it is ill-suited for decoding events with short inter-stimulus intervals. In order to overcome this issue, we propose a novel framework that separates the spatial and temporal components of the prediction by decoding the fMRI time-series continuously, i.e. scan-by-scan. The stimulation events can then be identified through a deconvolution of the reconstructed time series. We show that this model performs as well as or better than standard approaches across several datasets, most notably in regimes with small inter-stimuli intervals (3 to 5s), while also offering predictions that are highly interpretable in the time domain. This opens the way toward analyzing datasets not normally thought of as suitable for decoding and makes it possible to run decoding on studies with reduced scan time.

Keywords: Functional magnetic resonance imaging, Classification analysis, MVPA, Decoding, Rapid event-related design

1. Introduction

The application of multivariate analysis techniques to fMRI datasets, aka *decoding*, has become a popular approach to probe the relationships between stimuli and brain activity [20, 24, 14]. The very nature of fMRI data makes it a challenging problem: relatively few samples (events or blocks corresponding to stimulus presentation) are available, in comparison with the high dimensionality –number of voxels– of each observation. This mismatch leads to the so-called *curse of dimensionality*: learning distributed patterns from few samples is a hard problem. The power of high-dimensional regression methods is thus needed to achieve high accuracy and return an interpretable discriminative pattern (see e.g. [5]). However, the sluggishness of the Blood-Oxygen-Level-dependent (BOLD) response observed in fMRI implies that the occurrence of brain activity is not synchronous with the presentation of stimuli, but delayed by approximately 6s and smooth in time [9]. For the sake of statistical analysis, a preliminary regression step is thus typically performed, so that pairs of stimulus events and associated brain response can be considered. This prior regression is simply carried out by the traditional General Linear Model (GLM) used in standard statistical analyses of fMRI [8].

Although it is the standard solution used by nearly all practitioners, this two-step approach is not optimal; in particular, the intermediate event-related brain response estimates are very noisy, limiting decoding accuracy. The reason is that, unlike traditional brain mapping settings in which all events from one condition end up being one single regressor, for decoding purpose, events are split into different regressors, resulting in a loss in design efficiency and high-variance estimates. This approach is also bound to perform poorly on event-related tasks using small inter-stimuli inter-

vals (ISIs): the overlap in the hemodynamic response functions (HRFs) coupled with the acquisition noise lead to ill-estimated event-related brain responses and harm subsequent classification accuracy. In this work, we investigate a novel inference scheme that swaps the two steps: we propose to perform the challenging and expensive estimation of the discriminative pattern in the time frame of the slow BOLD response. For this we substitute the standard classification problem with one where the prediction target is a model of BOLD activity that includes the hemodynamic delay and blurring: we call this approach *time-domain fMRI decoding*. By nature, this approach uses the fMRI data to build a predictor of the convolved stimulus function; then a second step is necessary to go from condition-specific time courses to the identification of events. Given the estimated time course of several conditions during an acquisition, we use a second predictive model to decide which stimulus was presented at a given time. The key point is that this learning problem is *easy*, as it boils down to selecting a deconvolution filter and applying a relatively simple selection mechanism.

The expected benefit of this approach is to remain tractable whenever the ISI is short (3s to 5s). The promise of decoding with shorter ISIs is to optimize scanning efficiency: it allows for either reducing scan duration, leading to cheaper acquisitions and less taxing on subject's attention, or alternatively for keeping scan durations unchanged and acquiring more data per experiment.

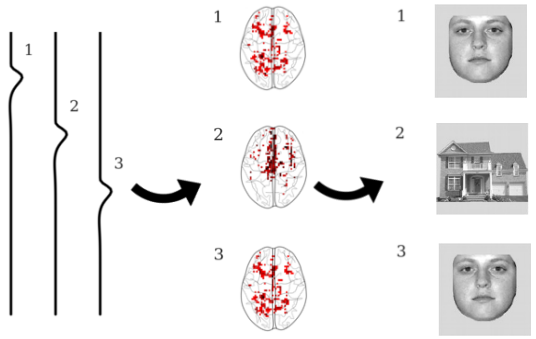


Figure 1: **Schema of the GLM approach to decoding**. A design matrix representing ideal task responses is used to derive event-specific activation estimates. These estimates are then used in a classification setting.

In the sequel, we describe the so-called *time-domain decoding* framework, and present experiments to compare it with state-of-the-art alternatives: the standard GLM-based regression, the so-called separate GLM approach and spatio-temporal analysis schemes. The fMRI datasets used for validation were chosen so as to represent a wide range of experimental settings, with block and event-related designs, the latter with ISIs ranging from 1.6 to 11.5s. We illustrate accuracy gains in these different settings. Before describing in detail the time-domain decoding approach and our experiments, we review state-of-the-art solutions.

2. Prior Work

Most decoding studies today are done fitting a first-level GLM regression: a design matrix $\mathbf{X}$ is created having as columns the timing of the experimental events, convolved with a canonical HRF model, and possibly additional columns to capture nuisance effects. Such an approach is illustrated in Figure 1. A crucial fact is that events corresponding to the same condition are disseminated into different columns, leading to poor (high-variance) per-trial activation estimates.

The activation coefficients are then estimated by solving the $\mathbf{X}\boldsymbol{\beta} = \mathbf{Y}$ regression problem, where $\mathbf{Y}$ are the BOLD data, written as a (time, voxel) array. The resulting least-squares estimates $\hat{\boldsymbol{\beta}}$ for activation coefficients have one value per voxel, hence they make up brain images, one image per event. Data classification is then performed by fitting a classifier to these activation maps: each activation image $\hat{\boldsymbol{\beta}}_i$ is associated with a label l_i, that indexes the cognitive condition corresponding to this event. Multivariate inference typically proceeds by estimating a function that predicts l given $\hat{\boldsymbol{\beta}}$: this function is a classifier (support vector machine, or logistic regression model) when the labels $(l_i)_{i=1..I}$ are discrete, or a regression function when the $(l_i)_{i=1..I}$ are continuous.

To summarize, this approach entails three estimation challenges:

- The one-event-per-column design is statistically inefficient [22].
- Curse of dimensionality: decoding is performed on brain-wide maps estimated based on a limited number of samples.

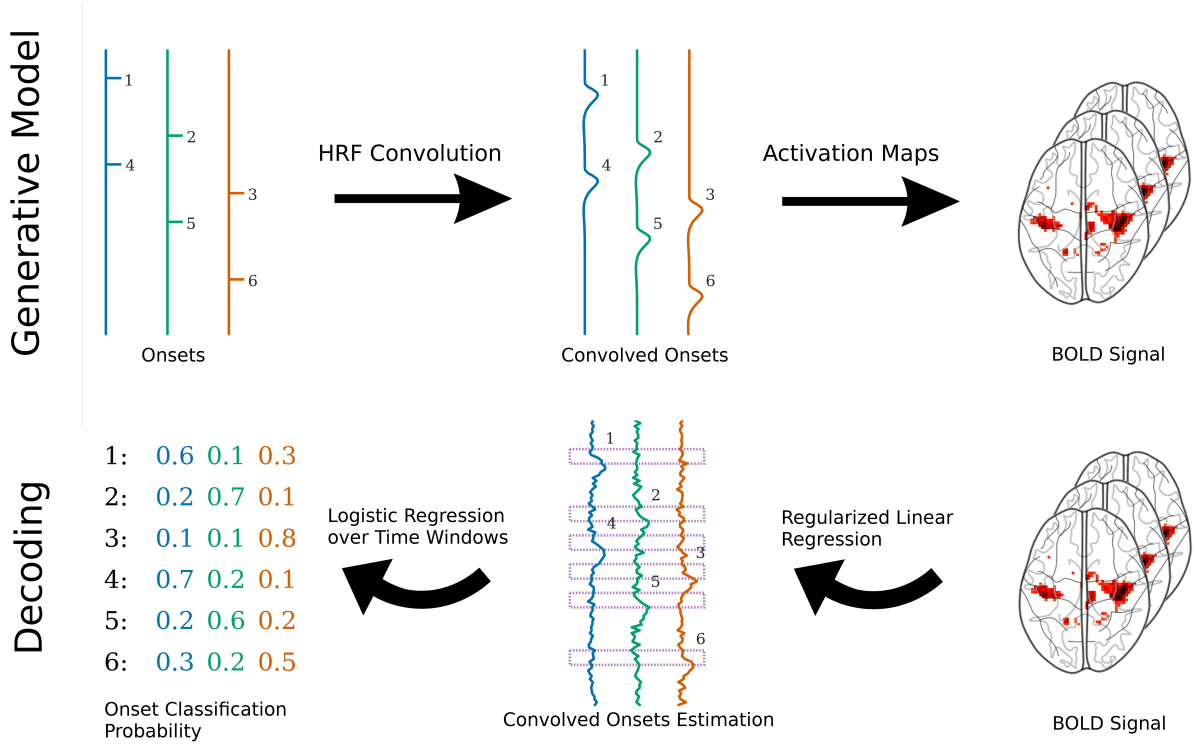


Figure 2: **Schema of the time-domain decoding model**. Straight arrows represent generative steps, while curved ones represent estimation steps.

- When trying to decode events with small ISIs, the regressors $(\mathbf{X}_i)_{i=1..I}$ used in the initial regression become highly correlated, thus rendering the first-level estimation ill-posed.

The so-called separate GLM or GLMs approach [19] has been proposed as a means of tackling the first and third issues: it is analogous to the GLM, only a separate design matrix $\mathbf{X}^{(i)}$ is built for deconvolving each trial i. These separate design matrices only have two columns (besides confounds): one for the stimulus regressor and one for the sum of all other regressors. The activation map for event i is obtained by solving $\mathbf{X}^{(i)}\boldsymbol{\beta}_i = \mathbf{Y}$, and the classification is then done through logistic regression on $(\boldsymbol{\beta}_i, l_i)_{i=1..I}$ as in the usual setting.

One thing that the GLM and GLMs approaches have in common is that they proceed by isolating temporal features in order to create activation maps. This is typically done by assuming a standard or canonical HRF model. Yet, extensions to data-driven approaches have been proposed in that framework [23][21], using finite impulse response filters. It should however be noted that these data-driven models require lots of stimulus occurrences, as they need to estimate regression coefficients in each voxel. Again, this types of model err on the large-variance side, given that a great number of coefficients are estimated per voxel.

Instead of isolating temporal features and decoding over activation maps, a different approach is to perform classification in the time domain. To capture the information from fMRI time series directly, the so-called Spatiotemporal SVM approach has been proposed [18], in which, for each event i, a vector $\mathbf{Y}_i$ is created by concatenating BOLD scans in a time-window following the stimulus onset. Classification is then performed by fitting a linear SVM over these concatenated vectors. While this approach nicely bypasses the prior specification of an HRF, it makes the problem worse regarding the "curse of dimensionality" (second issue outlined above): the number of features in the BOLD sig-

nal is multiplied by the length of the time window, thus rendering the classifier fitting problem even more ill-posed.

3. Time-domain decoding: a two-step approach

3.1. Motivation

One way of overcoming the dimensionality and efficiency issues with decoding procedures is to swap the spatial and the temporal estimation problems, by applying the time-lagged analysis in a low-dimensional space. This is the basis for the method proposed here.

3.2. Model

The Time-domain decoding method (3.2) first recovers the class-specific BOLD time courses to then assigns a class label to events. It comprises two steps:

1. Regression of the class-by-class convolved events time-series;
2. Classification of the stimulus occurrences based on time windows extracted from the regressed time-series.

More formally, let $\mathbf{X}$ be the design matrix of size (number of scans $\times$ number of stimulus classes), containing in each column the time-series for each stimulus class already convolved by an HRF model. Importantly, all the events of any given class are gathered into a single regressor. Let us note the BOLD signal matrix by $\mathbf{Y}$. It is assumed that high-pass filtering and motion parameter regression have been performed as a preprocessing step on Y. The data can be divided (across sessions) into train and test subsets $\mathbf{Y}_{train}$ and $\mathbf{Y}_{test}$, the corresponding design matrices being $\mathbf{X}_{train}$ and $\mathbf{X}_{test}$; the spatial step consists in solving a regularized regression problem (written here with a Ridge penalty):

$$\begin{aligned}\hat{\mathbf{B}} &= \underset{\mathbf{B}}{\operatorname{argmin}} \left\{ \|\mathbf{X}_{train} - \mathbf{Y}_{train}\mathbf{B}\|^2 + \lambda \|\mathbf{B}\|^2 \right\}, \\ \widetilde{\mathbf{X}}_{test} &= \mathbf{Y}_{test}\hat{\mathbf{B}}, \end{aligned} \tag{1}$$

where λ is a positive scalar to be specified; in the experiments described in this paper, it is set by nested cross-validation (for more details, see Annex A). Note that $\mathbf{B}$ is a $(n_{voxels} \times n_{conditions})$ matrix similar to a standard parameter matrix.

Once an estimate $\widetilde{\mathbf{X}}_{test}$ is obtained, the temporal step classifies each onset using a time window of t scans. Thus, if we denote by $\widetilde{\boldsymbol{x}}_{test[i:i+t]}$ the vector formed by concatenating the rows from i to $i+t$ of $\widetilde{\mathbf{X}}_{test}$, the temporal step consists in determining, for each onset time i, its corresponding class label $\boldsymbol{l}_i$. This is done by multiclass logistic regression:

$$\boldsymbol{l}_i = \underset{\boldsymbol{c_j}, j \in [\![1, n_{classes}]\!]}{\operatorname{argmax}} \left\{ \operatorname{logit}(< \hat{\boldsymbol{w}}_{c_j}, \widetilde{\boldsymbol{x}}_{test[i:i+t]} > + \hat{\boldsymbol{b}}_{c_j}) \right\}, \tag{2}$$

where $\hat{\boldsymbol{w}}_{c_j}$ and $\hat{\boldsymbol{b}}_{c_j}$ are computed for each class c_j by maximum likelihood estimation. Note that the number of weight coefficients estimated in this step is only t times the number $n_{classes}$ of classes.

This can be seen in the following manner: the first step (eq. 1) handles the decoding problem as a regression task, in which the strength of the hemodynamic response to each class is fitted by a regularized linear regression model over the BOLD signal on a scan-by-scan basis. Note that this first step relies on a fixed HRF model. In practice we chose the canonical double-gamma function of SPM.

The second, temporal step (see eq. 2 and figure 3) extracts from all classes' predicted time-series a time-window for each onset, and feeds them to a logistic regression model that works as a deconvolution filter for classifying the onset. The time-window length is defined heuristically using the canonical time dynamics of the HRF.

4. Experiments

4.1. Models

We compared the following classification methods: GLM, GLMs, Spatiotemporal SVM and Time-domain decoding described in Sections 2 and 3. Comparisons are presented on 4 datasets. We provide also simulations in appendix (section 7) that reproduce the results on fMRI datasets.

For the time-based models (Spatiotemporal SVM, Time-domain decoding), the time windows were chosen as the closest possible interval to the 2-8s range for event-related design datasets to ensure a good fit of the peak of the canonical HRF, and as the length of the block for datasets with a block design; we present in the annex some data to

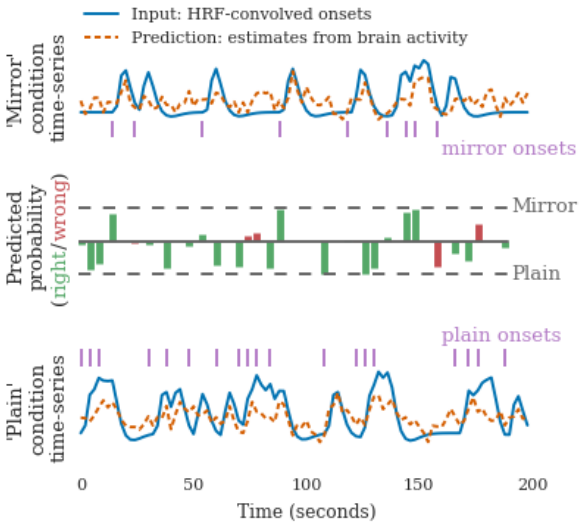


Figure 3: **Illustration of time-domain decoding on real data** using the *Mirror-reversed text* dataset (see Section 4.3): starting from the BOLD signal, the time-series are estimated for the two classes ('Plain' and 'Mirror') that are the two main conditions. Next, at each stimulus onset, a logistic regression is applied to a time-window in order to classify it. The class probabilities for each onset are shown in the middle graph, with a baseline of 50% at chance; the direction shows which class is predicted as being more likely; the length of the bars encodes the decision confidence and their color represents the correctness of that prediction.

discuss the choice of time-window in model performance in a post-hoc experiment (Fig. 11).

4.2. Simulation study

We performed a simulation study to assess the impact of ISI on model performance in a controlled setting. We generated data using a model of the form $\mathbf{Y} = \mathbf{X}\boldsymbol{\beta} + \boldsymbol{\epsilon}$:

- The design matrix $\mathbf{X}$, of shape (number of scans $\times$ 2), was created by convolving stimuli randomly assigned to one of two classes, separated by the ISI chosen for the session. As in the real data studies, we use the HRF model of [9]. We chose a number of scans of 1000 for the ISI=5s condition (as an approximation for the concatenation of the number of scans across all runs in a real experiment), and then adjusted for the other classes so as to have balanced number of stimuli for each ISI length (thus yielding 800 scans for ISI=4s and 600 for ISI=3s). We can note that, given a TR of 2s, this would correspond to 33 minutes of scanning time for the 5s ISI, 26 minutes for the 4s ISI and 20 minutes for the 3s one (not counting resting intervals).

- For the two classes $\boldsymbol{\beta}$, we created the activation maps, a matrix of shape (2×10000), by drawing their coefficients from a multivariate normal distribution $\mathcal{N}(3, \sigma_\beta^2 I)$, where σ_β was set to .5. We chose both the mean and the standard deviation as in the simulation study in [19]. The number 10000 for the features was chosen so as to reflect the number of features chosen by ANOVA variable selection in the real data studies.

- We generated the noise $\boldsymbol{\epsilon}$, a matrix of shape (number of scans $\times$ 10000), from an i.i.d. normal distribution $\mathcal{N}(0, \sigma_\epsilon^2)$, where σ_ϵ was set to 1.6, again as in [19]. We then smoothed the noise both temporally and spatially using a Gaussian filter with unit standard deviation, which led to an average standard deviation of approximately 0.72 for $\boldsymbol{\epsilon}$ across simulations.

We set the TR to 2 seconds and test ISIs of 3, 4, and 5 seconds. Results for activation map correlations of 0, 0.3 and 0.6 are also shown in 7. For each ISI and correlation, we run 100 simulations, in which we generate both a train and a test set with the procedure described above.

4.3. Real data

In order to probe different timing intervals and decoding complexity levels, we considered 4 datasets:

- The *Haxby* dataset [13] yields a study of face and object representations in human ventral temporal cortex, with 6 subjects and 12 runs per subject. Stimuli consisted of greyscale images from eight different classes: faces, cats, houses, chairs, scissors, shoes, bottles and scrambled images, and we considered the 8-class classification problem. Images for each class were shown during 24s, followed by 12s of rest; TR=2.5s, ISI=36s;

- The *Mirror-reversed text* [15] dataset yields a study of the neural basis of task-switching, with 14 subjects and 6 runs per subject. Stimuli were words shown in either plain or mirror-reversed fashion, coupled with a semantic classification task. The design is event-related, with TR=2s and ISI=4-11.5s.

- The *Textures* dataset [6] yields a study of responses to textures along different regions of interest in the brain, with 4 subjects undergoing a total of 7 acquisitions (3 subjects having gone through two acquisitions), with 6 runs per acquisition. Stimuli were greyscale texture images from 6 different classes in the UIUC texture database [17], appearing during three flashes of 200ms, separated by 200ms grey screen, so that an event duration is 1s. These images were shown in pairs separated by 4s followed by a probe after which the subject had to decide which of the first two the third image shown was extracted from. The design is event-related design, with TR=2.4s and ISI=4-8s.

- The *Temporal tuning* dataset [10] yields a study of rate-dependence of neural responses, with 11 subjects and 12 runs per subject. Greyscale images of faces and houses were shown in alternating fashion during 20s blocks, followed by 10s of rest. The design is event-related, with TR=1.5s and ISI=1.6, 3.2 or 4.8s.

Performance was analyzed in a within-subject setting. The cross-validation method used was Leave-one-session-out for *Mirror-reversed text* and *Textures*, and Leave-two-sessions-out for *Haxby* and *Temporal tuning* (based on the heuristic of having approximately 20% of the data in the test set). Cross-validation on the *Temporal tuning* dataset also followed a class-rebalancing scheme described in detail in section 7. We used classification accuracy of the events as the evaluation metric. One score was computed per cross-validation step, and significance of mean accuracy difference between methods was tested using paired t-tests.

4.4. Implementation

The analyses were performed in Python using the module Nilearn version 0.2.6, with Scikit-learn version 0.17.1 and Numpy version 1.11.3. Statistical analyses were performed using Nistats version 0.1.0, using the SPM model for the HRF. Plots were created using Matplotlib version 1.5.1 and Seaborn version 0.7.1. An implementation of the analysis for the public *Haxby* dataset can be found at `https://github.com/joaoloula/time_decoding`.

5. Results

Fig. 4 shows the results of the simulation study with no correlation between activation maps. The simulation suggests greater performance of the Time-domain decoding method over alternatives for this controlled environment, particularly for small ISIs.

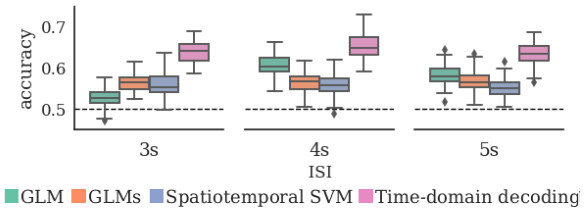


Figure 4: **Simulation study showing the prediction accuracy for varying ISIs**. The dotted line represents the chance level (50%).

For the real data, the relative accuracies for all methods on the *Haxby*, *Mirror-reversed text* and *Texture decoding* datasets are presented in Figure 5.

- On the *Haxby* dataset, Time-domain decoding outperforms all other methods ($p < 10^{-10}$, uncorrected), showing that it does well in traditional block designs. We can also see that this is the dataset in which Spatiotemporal SVM has its worst relative performance : th is is most likely an effect of the curse of dimensionality, given that the time-window is largest in this dataset. The (across methods) mean accuracy is 49% and the chance level is 12.5%;

- On the *Mirror-reversed text* dataset, Time-domain decoding outperforms GLM ($p < 10^{-9}$, uncorrected), and is outperformed by Spatiotemporal SVM ($p < 10^{-3}$, uncorrected).

The mean accuracy is 76% and the chance level is 50%.

- On the *Texture decoding* dataset, Time-domain decoding outperforms GLM ($p < 10^{-8}$, uncorrected) and Spatiotemporal SVM ($p < 10^{-7}$, uncorrected) and is outperformed by GLMs ($p < 0.05$, uncorrected). The mean accuracy is 43% and the chance level is 16.7%.

In Figure 6, we give results on the *Temporal tuning* dataset: the accuracies for each method, obtained through the cross-validation procedure described in 7, are shown separately according to the test-set ISI, which can be of 1.6, 3.2 or 4.8 seconds.

- When ISI=1.6 seconds, no method performs significantly better than chance;
- When ISI=3.2 seconds, Time-domain decoding outperforms GLMs ($p < 10^{-6}$, uncorrected);
- When ISI=4.8 seconds, Time-domain decoding significantly outperforms GLM ($p < 0.05$, uncorrected), GLMs ($p < 10^{-4}$, uncorrected) and Spatiotemporal SVM ($p < 10^{-6}$, uncorrected).

The fact that GLMs is outperformed by GLM for small ISIs in this dataset is most likely a consequence of high inter-trial variability (see [1]). The simulations results presented in section 7 confirm the superiority of the time-domain decoding method for ISIs of 3s to 5s.

Finally, Figure 7 shows the activation maps obtained for all methods on the Face vs. House task on the *Haxby* dataset. We can see that the maps for the four methods are highly similar: this indicates that the prediction problem as posed by the Time-domain decoding method still yields meaningful brain maps.

6. Discussion

Our experiments on the simulated data and four different real datasets establish clearly that the Time-domain decoding method performs as well as or better than state-of-the-art approaches. It does so in spite of the differences between datasets with respect to their timing characteristics. Put differently, this means that this approach is a safe default choice for the sake of decoding performance.

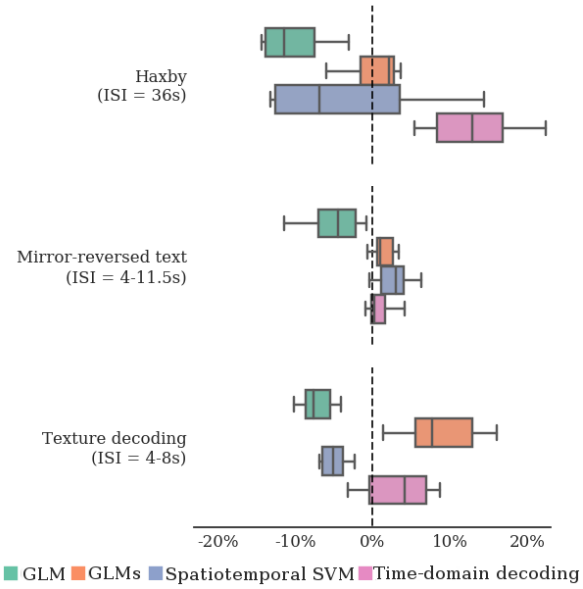


Figure 5: **Subject-by-subject accuracy comparison between GLM, GLMs, Spatiotemporal SVM and Time-domain decoding** across all datasets. Only the per-fold accuracy difference between methods is plotted in these figures: the dotted line represents the per-fold mean performance across methods. The mean accuracies (chance levels) are respectively: 49% (12.5%) on Haxby, 76% (50%) on Mirror-reversed text and 43% (16.7%) on Textures.

The results on the Temporal Tuning dataset are of particular interest: though extremely small ISIs degrade the performance of all methods to chance level, with an ISIl of 3.2s, Time-domain decoding outperforms GLMs and Spatiotemporal SVM, and at 4.8s it outperforms all other methods. With respect to the Spatiotemporal SVM, in particular, we confirm that Time-domain decoding does not suffer from the additional ill-posedness inherent to the strategy that augments the dimension of the input space. The direct implication of this result is that decoding becomes usable for ISIs as low as 3-4s, without jeopardizing too much prediction accuracy. This is thus a useful contribution toward cheaper, less demanding experiments for participants and opens the possibility to re-analyze existing datasets that have not been designed for decoding purposes.

More in detail, the first step of the Time-domain decoding uses a pre-defined HRF at learning time (to form the time courses used to train the spatial decoder), while the second step does not rely on a

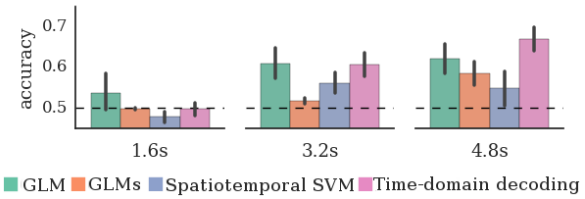


Figure 6: Accuracy comparison between GLM, GLMs, Spatiotemporal SVM and Time-domain decoding across different ISIs on the *Temporal tuning* dataset.

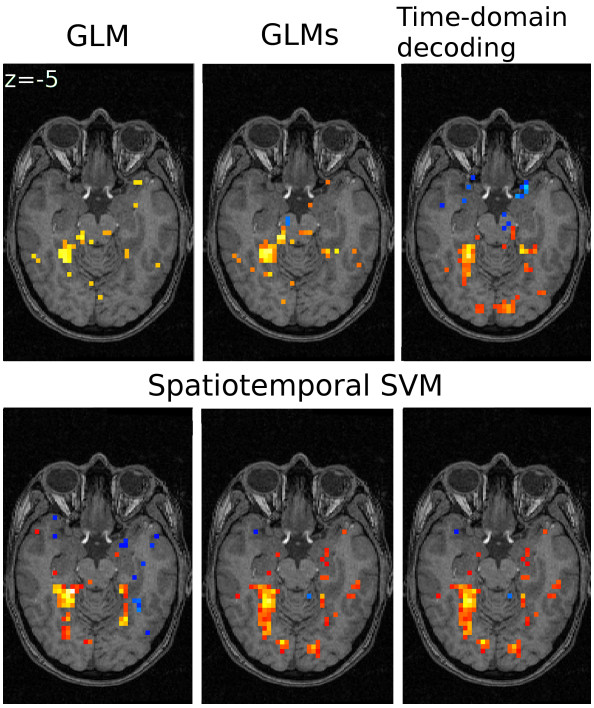


Figure 7: **Activation maps for Face vs. House classes on the *Haxby* dataset** for subject 2 for all models, shown at a 99.5% percentage threshold of signal strength. Since Spatiotemporal SVM produces a map for each scan in the time window (10 in the case of the *Haxby* dataset), we present only 3 maps corresponding to the timepoints of 2, 4 and 6s. We note the similarity between the maps, namely in the activation of the Fusiform Face Area: this seems to indicate that the regression problem posed by Logistic Regression yields meaningful maps.

temporal model. Our experience is that the procedure is robust to the choice of convolution model amongst canonical options e.g. using the canonical SPM response [7]. On the other hand, the deconvolution is a very sensitive step: in particular we have considered using model-based deconvolution instead of temporal decoding as in Eq. 2 –actually inverting a canonical generative model of the data– but this systematically lead to poorer predictions. The Time-domain decoding is also of a different nature than other spatio-temporal methods such as the one presented in [16], that tackle the question of decoding without timing information as opposed to that of separating stimuli in rapid succession. Notably, the *total activation* approach leverages a prior on neural events timing (minimization of the discontinuities) that is not used here.

It is worth noting that the Time-domain decoding method is distinct from HRF estimation approaches, as it does not aim at recovering the actual HRF that couples neural reponses with BOLD signals (an HRF model is used only in the first step to set X_{train}): no physiological model is estimated, and instead, the convolution filter is handled in the temporal step as a nuisance factor that simply needs to be inverted. In this respect, the presented approach is a discriminative rather than a generative model. In particular, while the HRF is likely to vary across brain regions [4, 21, 2], the temporal deconvolution performed in this work is an abstract, location-free filter estimate. Critically, it may not correspond to the local signal model of any brain region, although it can be interpreted as an inverse filter of the average HRF.

Aside from performance, two advantages of the Time-domain decoding model are worth pointing out:

- The decoupling of the spatial and temporal steps makes the method modular, and therefore particularly well-adapted to the substitution of richer models at each step. Possibilities include performing the spatial regression step using a spatially-regularized Graph-Net [12] or TV-L1 model [3] [11], or performing the low-dimensional deconvolution with another classifier such as Random Forest.

- The introduction of time-series for each class as an intermediary prediction step provides useful time-domain interpretability: a misclassification can be traced back to the time-series (see for example Figure 3). Notably, brain activations for each class can be tracked throughout scan times, allowing one to observe the ef-

fects of rivaling class-specific time courses and signal strength on decoding performance.

Finally, it should be noted that the use of Time-domain decoding can enhance the consequence of bad experimental design: if there exists a time-domain dependency between the occurrence of the different classes, it is possible that the model will capture those characteristics rather than meaningful cognitive features. Its use can therefore only be recommended on datasets with properly randomized events.

7. Conclusion

We presented the Time-domain decoding method for multivariate fMRI data decoding, which allows for efficient decoding with smaller ISIs than the state of the art, and is flexible to HRF variations. By design, it avoids the computational burden associated with time embedding approach used so far in the so-called spatio-temporal hrf model. The method is modular in nature, with weakly coupled spatial and temporal steps, and offers interpretability in the time domain. It has been shown to perform robustly on four different datasets, and to outperform alternatives in a short-ISI dataset. Code implementing the method as well as examples on the *Haxby* dataset can be found at `https://github.com/Joaoloula/time_decoding`.

Acknowledgements. We are grateful to Baptiste Gauthier for allowing us to use the Temporal Tuning dataset in the experiments presented here. This project has received funding from the European Union's Horizon 2020 Framework Programme for Research and Innovation under Grant Agreement No 720270 (Human Brain Project SGA1).

References

[1] H. Abdulrahman and R. N. Henson. Effect of trial-to-trial variability on optimal event-related fmri design: Implications for beta-series correlation and multi-voxel pattern analysis. *NeuroImage*, 125:756–766, 2016.

[2] S. Badillo, T. Vincent, and P. Ciuciu. Group-level impacts of within- and between-subject hemodynamic variability in fmri. *NeuroImage*, 82:433–448, Nov. 2013. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2013.05.100.

[3] L. Baldassarre, J. Mourao-Miranda, and M. Pontil. Structured sparsity models for brain decoding from fmri data. In *Proceedings of the 2012 Second International Workshop on Pattern Recognition in NeuroImaging*, PRNI '12, pages 5–8, Washington, DC, USA, 2012. IEEE Computer Society. ISBN 978-0-7695-4765-7. doi: 10.1109/PRNI.2012.31. URL `http://dx.doi.org/10.1109/PRNI.2012.31`.

[4] P. Ciuciu, J.-B. Poline, G. Marrelec, J. Idier, C. Pallier, and H. Benali. Unsupervised robust nonparametric estimation of the hemodynamic response function for any fmri experiment. *IEEE transactions on medical imaging*, 22:1235–1251, Oct. 2003. ISSN 0278-0062. doi: 10.1109/TMI.2003.817759.

[5] D. D. Cox and R. L. Savoy. Functional magnetic resonance imaging (fmri)brain reading: detecting and classifying distributed patterns of fmri activity in human visual cortex. *Neuroimage*, 19(2):261–270, 2003.

[6] M. Eickenberg, F. Pedregosa, S. Mehdi, A. Gramfort, and B. Thirion. Second order scattering descriptors predict fmri activity due to visual textures.

[7] R. S. Frackowiak, K. J. Friston, C. D. Frith, R. J. Dolan, C. J. Price, S. Zeki, J. T. Ashburner, and W. D. Penny. *Human brain function*. Academic press, 2004.

[8] K. J. Friston, A. P. Holmes, K. J. Worsley, J.-P. Poline, C. D. Frith, and R. S. J. Frackowiak. Statistical parametric maps in functional imaging: A general linear approach. *Human Brain Mapping*, 2(4):189–210, 1994. ISSN 1097-0193. doi: 10.1002/hbm.460020402. URL `http://dx.doi.org/10.1002/hbm.460020402`.

[9] K. J. Friston, P. Jezzard, and R. Turner. Analysis of functional mri time-series. *Human brain mapping*, 1(2):153–171, 1994.

[10] B. Gauthier, E. Eger, G. Hesselmann, A.-L. Giraud, and A. Kleinschmidt. Temporal tuning properties along the human ventral visual stream. *The Journal of neuroscience : the official journal of the Society for Neuroscience*, 32:14433–14441, Oct. 2012. ISSN 1529-2401. doi: 10.1523/JNEUROSCI.2467-12.2012.

[11] A. Gramfort, B. Thirion, and G. Varoquaux. Identifying predictive regions from fmri with tv-l1 prior. In *Proceedings of the 2013 International Workshop on Pattern Recognition in Neuroimaging*, PRNI '13, pages 17–20, Washington, DC, USA, 2013. IEEE Computer Society. ISBN 978-0-7695-5061-9. doi: 10.1109/PRNI.2013.14. URL `http://dx.doi.org/10.1109/PRNI.2013.14`.

[12] L. Grosenick, B. Klingenberg, K. Katovich, B. Knutson, and J. E. Taylor. Interpretable whole-brain prediction analysis with graphnet. *NeuroImage*, 72:304–321, May 2013. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2012.12.062.

[13] J. V. Haxby, M. I. Gobbini, M. L. Furey, A. Ishai, J. L. Schouten, and P. Pietrini. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science (New York, N.Y.)*, 293:2425–2430, Sept. 2001. ISSN 0036-8075. doi: 10.1126/science.1063736.

[14] J.-D. Haynes and G. Rees. Decoding mental states from brain activity in humans. *Nature reviews. Neuroscience*, 7:523–534, July 2006. ISSN 1471-003X. doi: 10.1038/nrn1931.

[15] K. Jimura, F. Cazalis, E. R. S. Stover, and R. A. Poldrack. The neural basis of task switching changes with skill acquisition. *Frontiers in human neuroscience*, 8:339, 2014. doi: 10.3389/fnhum.2014.00339.

[16] F. I. Karahanolu, C. Caballero-Gaudes, F. Lazeyras, and D. Van de Ville. Total activation: fmri deconvolution through spatio-temporal regularization. *NeuroImage*, 73:121–134, June 2013. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2013.01.067.

[17] S. Lazebnik, C. Schmid, and J. Ponce. A sparse texture representation using local affine regions. *IEEE transactions on pattern analysis and machine intelligence*, 27:1265–1278, Aug. 2005. ISSN 0162-8828. doi: 10.1109/TPAMI.2005.151.

[18] J. Mouro-Miranda, K. J. Friston, and M. Brammer. Dynamic discrimination analysis: a spatial-temporal svm. *NeuroImage*, 36:88–99, May 2007. ISSN 1053-8119. doi: 10.1016/j.neuroimage.2007.02.020.

[19] J. A. Mumford, B. O. Turner, F. G. Ashby, and R. A. Poldrack. Deconvolving bold activation in event-related designs for multivoxel pattern classification analyses. *NeuroImage*, 59:2636–2643, Feb. 2012. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2011.08.076.

[20] K. A. Norman, S. M. Polyn, G. J. Detre, and J. V. Haxby. Beyond mind-reading: multi-voxel pattern analysis of fMRI data. *Trends in Cognitive Sciences*, 10, 2006.

[21] F. Pedregosa, M. Eickenberg, P. Ciuciu, B. Thirion, and A. Gramfort. Data-driven hrf estimation for encoding and decoding models. *NeuroImage*, 104:209–220, Jan. 2015. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2014.09.060.

[22] S. Smith, M. Jenkinson, C. Beckmann, K. Miller, and M. Woolrich. Meaningful design and contrast estimability in fmri. *Neuroimage*, 34(1):127–136, 2007.

[23] B. O. Turner, J. A. Mumford, R. A. Poldrack, and F. G. Ashby. Spatiotemporal activity estimation for multivoxel pattern analysis with rapid event-related designs. *NeuroImage*, 62:1429–1438, Sept. 2012. ISSN 1095-9572. doi: 10.1016/j.neuroimage.2012.05.057.

[24] G. Varoquaux and B. Thirion. How machine learning is shaping cognitive neuroimaging. *GigaScience*, 3, 2014.

Annex A: Cross-validation scheme for the Temporal tuning dataset

The Temporal tuning dataset contains stimuli with ISIs of 4.8, 3,2 and 1.6 seconds. The experiment design, however, makes it so that these stimuli are not balanced: trials occur in blocks in which "Face" and "House" conditions are alternated with a given rate, making small ISI trials more numerous than larger ISI ones, as seen in Figure 8. To avoid biases introduced by the different number of samples, we subsampled the most frequent categories in order to achieve a balance in the number of examples for each ISI.

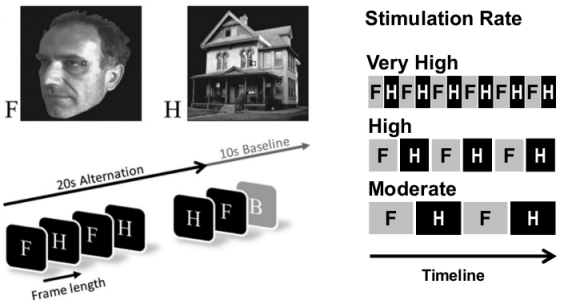


Figure 8: **Event structure of the Temporal tuning dataset**. Images of faces and houses were shown in alternation during continuous 20s blocks separated by 10s of rest. The ISI for each of these blocks was set to either 4.8, 3.2 or 1.6 seconds. It should be observed that this leads to an overabundance of trials with small ISI with respect to larger ISI ones: this makes rebalancing in the cross-validation necessary (see main text). Adapted from [10].

With these considerations, the cross-validation procedure performed was the following:

1. Different subsets of the 1.6s and 3.2s ISI stimuli are created, each containing the same number of examples as the set of 4.8s stimuli (which is the smallest group of the three, with 4 stimuli per block);
2. For each of these subsets, 20% of the data are left out in the validation set; the rest constitutes the decoding set. On the decoding set, a nested cross-validation loop is used to define the Ridge regression constant for the Logistic Regression model, as visualized in Figure 9.

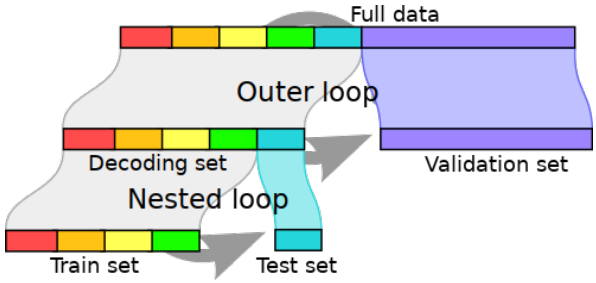


Figure 9: **Nested cross-validation procedure for the Logistic Regression model**. For each of the balanced subsets of the data, an inner CV loop is used to determine the Ridge hyperparameter while an outer one is used to assess model performance for each ISI.

Annex B: Simulation Study

In the context of the simulation study described in section 4.2, we also checked the influence of the correlation between activation maps on model performance. The setting is the same as in section 4.2, but now the distribution from which the activation maps are drawn has a non-trivial correlation structure $\sigma^2_{beta}(I+Cor)$, where $Cor_{i,j} = c$ if the features i and j correspond to the same voxel across the two maps and $Cor_{i,j} = 0$ otherwise. The value of c was made to vary between 0, 0.3 and 0.6, and for each ISI-correlation pair 100 simulations were run. The results are shown in fig 10.

We observe that, while higher correlation decreases performance across all methods (as expected), it does so in a non-homogeneous way: notably, while tests with low correlation show that Time-domain decoding outperforms all other methods, with a correlation of 0.6 the performance is mostly uniform across the four models. This indicates that Time-domain decoding does not address the issue of ill-posed spatial pattern estimation.

Annex C: Time window length analysis

In order to study the influence of the time-window length parameter on the performance of the temporal step of the Time-domain decoding method (as described in section 3) we performed tests on the Haxby dataset using 5 different window lengths: 2.5, 5, 7.5, 10 and 20 seconds, the first 4 centered around the canonical peak for the HRF (around 5 to 6 seconds after the stimulus onset),

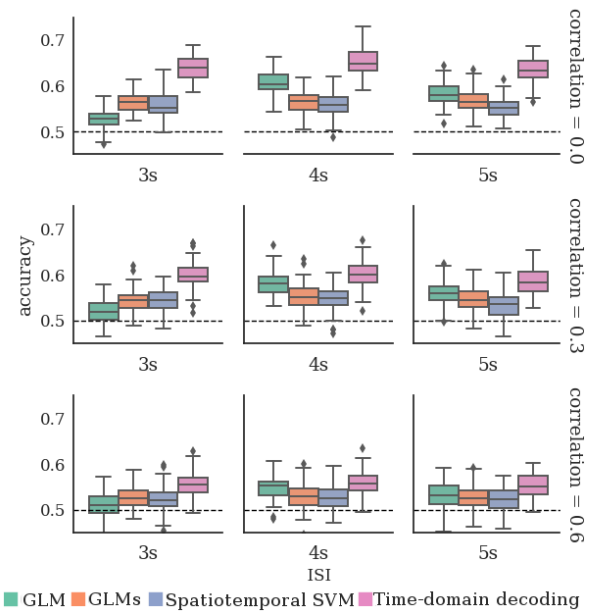


Figure 10: **Prediction accuracy for varying ISIs and between-map correlations (simulated data)** . The dotted line represents the chance level (50%). Correlations degrade the accuracy obtained, but perserves the relative performance of the methods.

and the 20s window beginning at the stimulus onset. These are equivalent respectively to one, two, three, four and eight scans: for the case with one scan, we performed classification by simply taking the label to be the one with maximum activation for that scan. The performance was measured across all 8 classes on the Haxby dataset, and the cross-validation procedure used was the same as the one in the previous experiments.

The accuracy metric across these different window lengths gives an indication of how variations in the time-window size affect decoding performance: as we can see, though performance is high across all windows, there is a high increase in the 7.5s window relative to the smaller ones. In particular, we notice that the decoder that only uses the maximum activation at one timestep for classification (the 2.5s window decoder) fails to achieve the performance of the time-domain decoding methods for longer time-windows, attesting the utility of the logistic regression as a time-domain deconvolution step. Given the notable ISI on the Haxby dataset, we see steady increase in accuracy with time-window length saturating around 10s, while doubling the length to 20s has almost no impact on performance.

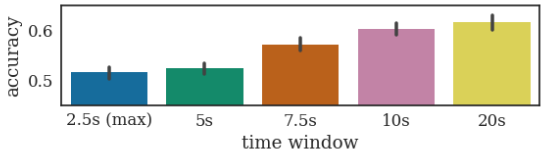


Figure 11: **Impact of the time-window length on the performance of the Time-domain decoding method** in the Haxby dataset. One can see that the jump from 5s to 7.5s lenght (2 to 3 scans) yields great improvement in performance, and that there is generally steady increase in accuracy with time-window length, saturating at around 10s. Recall that chance is $1/8 = .125$.