NeuroImage xxx (2017) 1–13

Contents lists available at ScienceDirect

NeuroImage

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

FReM – Scalable and stable decoding with fast regularized ensemble of models

Andrés Hoyos-Idrobo [a,b,*], Gaël Varoquaux [a,b], Yannick Schwartz [a,b], Bertrand Thirion [a,b]

[a] Parietal project-team, INRIA, Saclay-île de, France
[b] CEA/Neurospin bât 145, 91191, Gif-Sur-Yvette, France

ARTICLE INFO

Keywords:
fMRI
Supervised learning
Decoding
Bagging
MVPA

ABSTRACT

Brain *decoding* relates behavior to brain activity through predictive models. These are also used to identify brain regions involved in the cognitive operations related to the observed behavior. Training such multivariate models is a high-dimensional statistical problem that calls for suitable priors. State of the art priors –eg small total-variation– enforce spatial structure on the maps to stabilize them and improve prediction. However, they come with a hefty computational cost. We build upon very fast dimension reduction with spatial structure and model ensembling to achieve decoders that are fast on large datasets and increase the stability of the predictions and the maps. Our approach, *fast regularized ensemble of models* (FReM), includes an implicit spatial regularization by using a voxel grouping with a fast clustering algorithm. In addition, it aggregates different estimators obtained across splits of a cross-validation loop, each time keeping the best possible model. Experiments on a large number of brain imaging datasets show that our combination of voxel clustering and model ensembling improves decoding maps stability and reduces the variance of prediction accuracy. Importantly, our method requires less samples than state-of-the-art methods to achieve a given level of prediction accuracy. Finally, FreM is much faster than other spatially-regularized methods and, in addition, it can better exploit parallel computing resources.

Introduction: decoding needs stability

Decoding models predict stimuli or behavior from brain images. These models have become a standard tool in neuroimaging data analysis (Haynes and Rees, 2006; Norman et al., 2006; Varoquaux and Thirion, 2014). In clinical applications, they can be used to perform diagnosis or prognosis (Fan et al., 2008; Demirci et al., 2008). They are also used as evidence of the link between distributed activity patterns and an observed behavior (Haxby et al., 2001). Additionally, decoding used on a large variety of cognitive processes grounds a form of reverse inference (Poldrack, 2011; Schwartz et al., 2013). An appeal of decoding procedures is that they avoid multiple voxel-wise test and perform an omnibus test: "Can one predict the behavioral outcome from brain activity?"

Identifying the brain activity patterns that drive prediction of behavior is crucial for brain mapping and understanding (Mourão-Miranda et al., 2005; Gramfort et al., 2013). However achieving reliable and stable decoder maps is challenging due to the dimensionality of the problem: the number of samples is small –hundreds or less– whereas the number of features is typically the number of voxels in the brain –up to hundreds of thousands. Linear models, e.g. linear support vector machines (SVM), are often used (Pereira et al., 2009), as they have shown a good performance in a small-sample regime. In addition, their classification/regression weights form brain maps used for interpretation of the discriminative pattern (Mourão-Miranda et al., 2005).

However, the high dimensionality of the problem leads to multiple weight maps yielding the same predictive power, and some form of regularization has to be applied (Hastie et al., 2000). In across-subject settings, complex spatial and sparse penalties such as total-variation (TV) (Michel et al., 2011; Baldassarre et al., 2012) and Graph-net (Grosenick et al., 2013) help the decoder to capture the important brain regions shared across subjects. TV and its variants are considered as the state-of-the-art regularizers for brain images, as they handle local correlations present in the data. The main drawback of spatially-structured sparsity as in TV and related penalties is their computational cost.

A much cheaper alternative to these structured estimators is to use spatially-constrained clustering algorithms to perform voxel grouping. In decoding, voxel grouping is often used as part of the pipeline for stability selection of correlated voxels (Varoquaux et al., 2012; Gramfort et al., 2012; Wang et al., 2015). Additionally, it helps to improve the conditioning of the estimation problem. However, voxel grouping introduces

* Corresponding author. Parietal project-team, INRIA, Saclay-île de, France.
E-mail address: ahoyosidrobo@gmail.com (A. Hoyos-Idrobo).

https://doi.org/10.1016/j.neuroimage.2017.10.005
Received 16 March 2017; Received in revised form 28 September 2017; Accepted 3 October 2017
Available online xxxx

high bias, as the patterns are constrained by the clusters shape.

One way to mitigate this bias is to use model aggregation or ensembling. These approaches have been used to reduce the variability of the output of the decoder (Kuncheva and Rodríguez, 2010; Kuncheva et al., 2010a; Zhou, 2012). The central idea is to build a decoder by averaging the output of several "good" models. In particular, averaging linear models boils down to averaging weight maps. One way to estimate multiple models is to use bootstrap resampling to generate different training sets to fit the decoder, and then aggregate them. This approach is known as *Bagging*[1] (Breiman, 1996). It is easy to run in parallel, training each model independently. Yet, naive application of bagging to neuroimaging data induces high computational cost as the data are high dimensional, and parameters have to be set by internal cross-validation.

Decoding calls not only for hyperparameter selection, but also for model validation. Both tasks require a measure of the predictive power of the decoder. In practice, one runs two cross-validation loops –one inside the other– where each loop assesses prediction accuracy respectively for model selection and validation. Thus, investigators often train the decoder many times. These repeated calculations entail computational costs that limit day-to-day work on standard workstations. This is particularly problematic for more advanced decoders such as those with spatial regularizations that are beneficial to neuroimaging data (e.g. Mohr et al., 2015; Michel et al., 2011; Grosenick et al., 2013). In the face of growing data size, to enable good validation and ease of use on most hardware, a good decoder should be sparing on computation resources.

Contributions. Here, we propose a fast scheme to train regularized ensembles of models, FReM. It reduces the variance of the weight maps of the decoder, while ensuring high prediction accuracy. The core of this approach is to average the estimator with the best predictive power per loop inside the nested cross-validation. To benefit from spatial regularization while keeping fast run times, we show how an optional voxel-clustering can be included in the ensembling, bringing stable spatial patterns. We perform a series of classification experiments on several MRI datasets to demonstrate that ensembling regularized models gives state-of-the-art decoders. In particular, we show that they compare favorably to existing decoders in terms of prediction performance, weight-map stability, and computation time.

Background and prior art

Brain decoding

In neuroimaging, a decoder is a predictive model that, given n brain images, fits an external variable $\mathbf{y}$. In practice, we arrange n observed brain images composed of p voxels in a matrix $\mathbf{X} \in \mathbb{R}^{n \times p}$. Linear predictive models, at the core of most decoders in neuroimaging, are then written (Hastie et al., 2009):

$$\mathbf{y} = f(\mathbf{X}\mathbf{w} + \boldsymbol{\varepsilon}), \tag{1}$$

where $\mathbf{y}$ denotes a target variable giving the experimental condition or health status of subjects, f represents the decision function in the classification; $\mathbf{w} \in \mathbb{R}^p$ denotes the weight vector/map, and $\boldsymbol{\varepsilon} \in \mathbb{R}^n$ is a random error term.

In spite of a recently growing effort on the accumulation of neuroimaging data (Poldrack and Gorgolewski, 2015), the number n of samples per-class remains in the order of a few hundreds, whereas p can be hundreds of thousands of voxels ($p \gg n$). In this high-dimensional setting, there are many equivalent solutions and some form of regularization or prior is necessary to restrict model complexity. A standard approach relies on solving the following optimization problem:

$$\widehat{\mathbf{w}}(\lambda) = \underset{\mathbf{w} \in \mathbb{R}^p}{argmin}\{\mathscr{L}(\mathbf{y}, \mathbf{X}; \mathbf{w}) + \lambda \Omega(\mathbf{w})\}, \quad \lambda > 0, \tag{2}$$

where $\mathscr{L}$ is a data-fidelity term, a loss function that measures the quality of the estimator (e.g. logistic or hinge loss); Ω denotes the penalty/regularization term, and λ is the parameter that controls the amount of regularization. Two of the most often used penalties are: *1)* the ℓ_2-norm, that penalizes large $\mathbf{w}$ coefficients, and yields non-sparse solutions; *2)* the ℓ_1-norm, that promotes a small number of non-zero $\mathbf{w}$ coefficients, and yields sparse solutions (Tibshirani, 1994).

Nevertheless, as neuroimaging data exhibit strong correlations between the columns of $\mathbf{X}$, the ℓ_1-penalty yields unstable solutions as it tends to arbitrarily select only one among the correlated variables (Yu, 2013; Varoquaux et al., 2012). One way to tackle this is the use of additional spatially-informed penalties as Graph-net (Grosenick et al., 2013) or TV (total variation) (Michel et al., 2011; Eickenberg et al., 2015).

Model validation and selection

In high-dimensional settings, the number of candidate models is much larger than the number of samples. Therefore, we use regularization to constrain the complexity of the solution, and this penalization is controlled by the λ regularization parameter. The ensuing problem is then to find an optimal value for λ (i.e. finding the best bias-variance trade-off), yielding a model that exploits the richness of the data. One typically uses the predictive power of the decoder to choose the right amount of regularization.

Hyperparameters selection. In general, the setting of the hyperparameter is a data-specific choice, as it is governed by the amount of data and their signal-to-noise-ratio (SNR). The most common approach to set it is to use cross-validation to measure the predictive power for various amounts of regularization and retain the value that maximizes the predictive power across several cross-validation folds (Varoquaux et al., 2017). To assess predictive power in addition, the standard scheme is *nested cross*-validation, that consists of two cross-validation loops run one inside the other: an outer loop is used to assess the predictive power of the decoder, and an inner/nested loop is used to set the hyperparameter(s) (see Fig. 1): the train set is used to fit the decoder, while the test and validation sets are used to measure its ability to generalize to new data.

In most of the non-parametric approaches to select a regularization parameter, a suitable and finite set of l hyperparameters, $\lambda \in [\lambda_1, \ldots, \lambda_l]$ is first defined. For each cross-validation fold, one fits the decoder with all hyperparameters, and measures their prediction error on the test set. Then, one choses the λ_i value that maximizes the predictive power across folds.

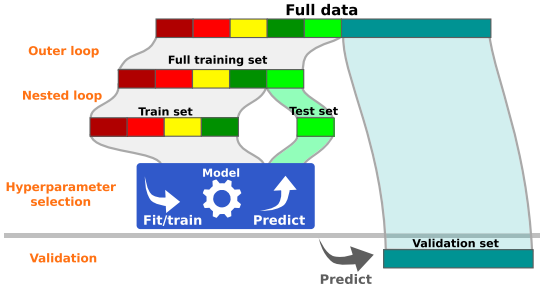

Fig. 1. Illustration of nested cross-validation: Two cross-validation loops are run one inside the other. The inner loop is used to set the hyperparameters, whereas the outer loop is used to assess the predictive power of the decoder.

[1] Bagging stands for Bootstrap aggregating.