NeuroImage 145 (2017) 166–179

Contents lists available at ScienceDirect

NeuroImage

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

Assessing and tuning brain decoders: Cross-validation, caveats, and guidelines

Gaël Varoquaux[a,b,*], Pradeep Reddy Raamana[c,d], Denis A. Engemann[b,e,f], Andrés Hoyos-Idrobo[a,b], Yannick Schwartz[a,b], Bertrand Thirion[a,b]

[a] Parietal project-team, INRIA Saclay-ile de France, France
[b] CEA/Neurospin bât 145, 91191 Gif-Sur-Yvette, France
[c] Rotman Research Institute, Baycrest Health Sciences, Toronto, ON, Canada M6A 2E1
[d] Department of Medical Biophysics, University of Toronto, Toronto, ON, Canada M5S 1A1
[e] Cognitive Neuroimaging Unit, INSERM, Université Paris-Sud and Université Paris-Saclay, 91191 Gif-sur-Yvette, France
[f] Neuropsychology & Neuroimaging team INSERM UMRS 975, Brain and Spine Institute (ICM), Paris

ARTICLE INFO

Keywords:
Cross-validation
Decoding
FMRI
Model selection
Sparse
Bagging
MVPA

ABSTRACT

Decoding, i.e. prediction from brain images or signals, calls for empirical evaluation of its predictive power. Such evaluation is achieved via cross-validation, a method also used to tune decoders' hyper-parameters. This paper is a review on cross-validation procedures for decoding in neuroimaging. It includes a didactic overview of the relevant theoretical considerations. Practical aspects are highlighted with an extensive empirical study of the common decoders in within- and across-subject predictions, on multiple datasets –anatomical and functional MRI and MEG– and simulations. Theory and experiments outline that the popular "leave-one-out" strategy leads to unstable and biased estimates, and a repeated random splits method should be preferred. Experiments outline the large error bars of cross-validation in neuroimaging settings: typical confidence intervals of 10%. Nested cross-validation can tune decoders' parameters while avoiding circularity bias. However we find that it can be favorable to use sane defaults, in particular for non-sparse decoders.

1. Introduction: decoding needs model evaluation

Decoding, i.e. predicting behavior or phenotypes from brain images or signals, has become a central tool in neuroimage data processing (Haynes and Rees, 2006; Haynes, 2015; Kamitani and Tong, 2005; Norman et al., 2006; Varoquaux and Thirion, 2014; Yarkoni and Westfall, 2016). In clinical applications, prediction opens the door to diagnosis or prognosis (Mouro-Miranda et al., 2005; Fu et al., 2008; Demirci et al., 2008). To study cognition, successful prediction is seen as evidence of a link between observed behavior and a brain region (Haxby et al., 2001) or a small fraction of the image (Kriegeskorte et al., 2006). Decoding power can test if an encoding model describes well multiple facets of stimuli (Mitchell et al., 2008; Naselaris et al., 2011). Prediction can be used to establish what specific brain functions are implied by observed activations (Schwartz et al., 2013; Poldrack et al., 2009). All these applications rely on measuring the predictive power of a decoder.

Assessing predictive power is difficult as it calls for characterizing the decoder on prospective data, rather than on the data at hand. Another challenge is that the decoder must often choose between many different estimates that give rise to the same prediction error on the data, when there are more features (voxels) than samples (brain images, trials, or subjects). For this choice, it relies on some form of regularization, that embodies a prior on the solution (Hastie et al., 2009). The amount of regularization is a parameter of the decoder that may require tuning. Choosing a decoder, or setting appropriately its internal parameters, are important questions for brain mapping, as these choice will not only condition the prediction performance of the decoder, but also the brain features that it highlights.

Measuring prediction accuracy is central to decoding, to assess a decoder, select one in various alternatives, or tune its parameters. The topic of this paper is cross-validation, the standard tool to measure predictive power and tune parameters in decoding. The first section is a primer on cross-validation giving the theoretical underpinnings and the current practice in neuroimaging. In the second section, we perform an extensive empirical study. This study shows that cross-validation results carry a large uncertainty, that cross-validation should be performed on full blocks of correlated data, and that repeated random splits should be preferred to leave-one-out. Results also yield guidelines for decoder parameter choice in terms of prediction

* Corresponding author at: Parietal project-team, INRIA Saclay-ile de France, France.

http://dx.doi.org/10.1016/j.neuroimage.2016.10.038
Received 3 August 2015; Accepted 24 October 2016
Available online 29 October 2016

performance and stability.

2. A primer on cross-validation

This section is a tutorial introduction to important concepts in cross validation for decoding from brain images.

2.1. Cross-validation: estimating predictive power

In neuroimaging, a decoder is a predictive model that, given brain images $\mathbf{X}$, infers an external variable $\mathbf{y}$. Typically, $\mathbf{y}$ is a categorical variable giving the experimental condition or the health status of subjects. The accuracy, or predictive power, of this model is the expected error on the prediction, formally:

$$\text{accuracy} = \mathbb{E}[\mathcal{E}(\mathbf{y}^{\text{pred}}, \mathbf{y}^{\text{ground truth}})] \quad (1)$$

where $\mathcal{E}$ is a measure of the error, most often[1] the fraction of instances for which $\mathbf{y}^{\text{pred}} \neq \mathbf{y}^{\text{ground truth}}$. Importantly, in Eq. (1), $\mathbb{E}$ denotes the *expectation*, *ie* the average error that the model would make on infinite amount of data generated from the same experimental process.

In decoding settings, the investigator has access to labeled data, *ie* brain images for which the variable to predict, $\mathbf{y}$, is known. These data are used to train the model, fitting the model parameters, and to estimate its predictive power. However, the same observations cannot be used for both. Indeed, it is much easier to find the correct labels for brain images that have been seen by the decoder than for unknown images.[2] The challenge is to measure the ability to *generalize* to new data.

The standard approach to measure predictive power is *cross-validation*: the available data is split into a *train set*, used to train the model, and a *test set*, unseen by the model during training and used to compute a prediction error (Fig. 1). Chapter 7 of Hastie et al. (2009) contains a reference on statistical aspects of cross-validation. Below, we detail important considerations in neuroimaging.

Independence of train and test sets. Cross-validation relies on independence between the train and test sets. With time-series, as in fMRI, the autocorrelation of brain signals and the temporal structure of the confounds imply that a time separation is needed to give truly independent observations. In addition, to give a meaningful estimate of prediction power, the test set should contain new samples displaying all confounding uncontrolled sources of variability. For instance, in multi-session data, it is harder to predict on a new session than to leave out part of each session and use these samples as a test set. However, generalization to new sessions is useful to capture actual invariant information. Similarly, for multi-subject data, predictions on new subjects give results that hold at the population level. However, a confound such as movement may correlate with the diagnostic status predicted. In such a case the amount of movement should be balanced between train and test set.

Sufficient test data. Large test sets are necessary to obtain sufficient power for the prediction error for each split of cross-validation. As the amount of data is limited, there is a balance to strike between achieving such large test sets and keeping enough training data to reach a good fit with the decoder. Indeed, theoretical results show that cross-validation has a negative bias on small dataset (Arlot and Celisse, 2010, Section 5.1) as it involves fitting models on a fraction of the data. On the other hand, large test sets decrease the variance of the estimated accuracy (Arlot and Celisse, 2010, Section 5.2). A good cross-validation strategy balances these two opposite effects. Neuroimaging papers often use

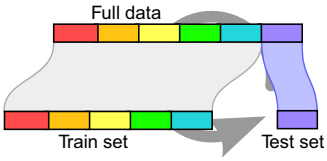

Fig. 1. Cross-validation: the data is split multiple times into a train set, used to train the model, and a test set, used to compute predictive power.

leave one out cross-validation, leaving out a single sample at each split. While this provides ample data for training, it maximizes test-set variance and does not yield stable estimates of predictive accuracy.[3] From a decision-theory standpoint, it is preferable to leave out 10% to 20% of the data, as in 10-fold cross-validation (Hastie et al., 2009, chap. 7.12; Breiman and Spector, 1992; Kohavi, 1995). Finally, it is also beneficial to increase the number of splits while keeping a given ratio between train and test set size. For this purpose k-fold can be replaced by strategies relying on repeated random splits of the data (sometimes called repeated learning-testing[4] (Arlot and Celisse, 2010) or *ShuffleSplit* (Pedregosa et al., 2011)). As discussed above, such splits should be consistent with the dependence structure across the observations (using *eg* a *LabelShuffleSplit*), or the training set could be stratified to avoid class imbalance (Raamana et al., 2015). In neuroimaging, good strategies often involve leaving out sessions or subjects.

2.2. Hyper-parameter selection

A necessary evil: one size does not fit all. In standard statistics, fitting a simple model on abundant data can be done without the tricky choice of a meta-parameter: all model parameters are estimated from the data, for instance with a maximum-likelihood criterion. However, in high-dimensional settings, when the number of model parameters is much larger than the sample size, some form of regularization is needed. Indeed, adjusting model parameters to best fit the data without restriction leads to *overfit*, *ie* fitting noise (Hastie et al., 2009, chap. 7). Some form of regularization or prior is then necessary to restrict model complexity, e.g. with low-dimensional PCA in discriminant analysis (Chen et al., 2006), or by selecting a small number of voxels with a sparse penalty (Yamashita, 2008; Carroll et al., 2009). If too much regularization is imposed, the ensuing models are too constrained by the prior, they *underfit*, *ie* they do not exploit the full richness of the data. Both underfitting and overfitting are detrimental to predictive power and to the estimation of model weights, the decoder maps. Choosing the amount of regularization is a typical bias-variance problem: erring on the side of variance leads to overfit, while too much bias leads to underfit. In general, the best tradeoff is a data-specific choice, governed by the statistical power of the prediction task: the amount of data and the signal-to-noise ratio.

Nested cross-validation. Choosing the right amount of regularization can improve the predictive power of a decoder and controls the appearance of the weight maps. The most common approach to set it is to use cross-validation to measure predictive power for various choices of regularization and to retain the value that maximizes predictive power. Importantly, with such a procedure, the amount of regularization becomes a parameter adjusted on data, and thus the predictive performance measured in the corresponding cross-validation loop is not a reliable assessment of the predictive performance of the model. The standard procedure is then to refit the model on the available data, and test its predictive performance on new data, called a *validation* set. Given a finite amount of data, a *nested cross-validation* procedure can

[1] For multi-class problems, where there is more than 2 categories in $\mathbf{y}$, or for unbalanced classes, a more elaborate choice is advisable, to distinguish misses and false detections for each class.

[2] A simple strategy that makes no errors on seen images is simply to store all these images during the training and, when asked to predict on an image, to look up the corresponding label in the store.

[3] One simple aspect of the shortcomings of small test sets is that they produce unbalanced dataset, in particular leave-one-out for which there is only one class represented in the test set.

[4] Also related is bootstrap CV, which may however duplicate samples inside the training set of the test set.