Advancing neuroimaging research with predictive multivariate pattern analysis

Yaroslav O. Halchenko and Michael Hanke

PyMVPA, a novel Python-based framework for multivariate pattern analysis, facilitates the application of statistical learning methods to neural data.

Nobel prize winner Eric Kandel wrote: “The task of neural science is to explain behavior in terms of the activities of the brain.” Unfortunately, the currently prevalent data analysis strategies do not aim at exploring behavior in terms of neural activity per se. Instead, the majority of methods primarily explore the data by performing mass-univariate hypothesis tests, searching for statistically significant excursions of the signal from a ‘no-effect’ baseline. Such approaches often rely on restrictive modeling assumptions: e.g. the forward model of a hemodynamic (blood circulation) response function in functional magnetic resonance imaging (fMRI). Because of this, they require pre-processing steps (spatial and temporal smearing, averaging, etc.) that necessarily ignore or obliterate some information embedded in the data. Furthermore, univariate modeling of the acquired signal in terms of behavioral factors neither considers present covariance and causal structure among distinct brain areas, nor does it account for the variance of the response patterns across trials.

In recent fMRI-based research, these limitations have led to a reconsideration of multivariate pattern analysis (MVPA) methods that had been introduced more than a decade ago in studies employing positron emission tomography. Enabled by recent advances in the field of statistical learning theory, some striking developments have attracted considerable interest throughout the neuroscience community. For instance, the application of regularized statistical classifiers (e.g., a support vector machine or SVM) allowed the reliable prediction of behavioral conditions based on full-brain fMRI data for each single trial. This reversal of the analysis strategy, where now aspects of behavior are modeled in terms of neural activity, represents a critical difference from previously established approaches (see Figure 1).

Despite the advantages and promise of these methods, various factors have delayed their adoption. Although a growing number of studies now employ statistical learning methods, the compressed verbal descriptions of the novel and rather complex analysis pipelines—coupled with the lack of a unified and flexible software framework—have hindered straightforward replication attempts. Nevertheless, replication—and hence validation of reported results by independent research groups—is essential for scientific progress.

Figure 1. Reversing the analysis flow: classical statistical parametric mapping (SPM) performs mass-univariate testing to localize hypothetical brain responses. In contrast, multivariate pattern analysis (MVPA) offers a direct quantifiable mapping from brain activity patterns onto behavioral states.

Continued on next page
To provide the neuroscience community with an adequate tool for the analysis of neural data using statistical learning methods, we have developed PyMVPA (Python MVPA\textsuperscript{11}). This is a free, open-source, and platform-agnostic project that utilizes the Python programming language. Python is a perfect choice because of its portability, its concise and descriptive syntax, and its ability to easily interface to low-level libraries and high-level scientific scripting environments, such as R.\textsuperscript{12} PyMVPA makes it easy to access data stored in standard data formats (e.g., NIfTI), to perform typical statistical learning procedures (such as training, testing, feature selection, and cross-validation without ‘peeking’ or ‘double-dipping’),\textsuperscript{13} while exploring the multitude of available learning methods and facilitating rapid development. It also makes it easy to allow contributions from any interested researcher.

We designed PyMVPA to offer a high-level programming interface that allows for a flexible combination of the provided building blocks to express complex analysis pipelines in just a few lines of code.\textsuperscript{12} This feature enables researchers to easily replicate existing studies, and to carry out novel non-standard analyses. Moreover, the descriptive power of human-readable, yet compact, source code opens the possibility of including the complete source code of a study as a supplemental material of a publication. (Mandatory code-inclusion research papers could tremendously expedite verification and adoption of novel analysis strategies.)

To demonstrate the power and applicability of the suggested analysis methodologies we\textsuperscript{14} analyzed data from four different neural modalities and accompanied the publication with the complete source code of all of them. Essentially the same workflow was used for all neural data modalities: basic preprocessing, training and testing (by cross-validation) of statistical classifiers, and the analysis of the trained classifiers sensitivities with respect to any given input dimension. Applied to extracellular recordings data (post-stimulus time histograms of spike counts) it was possible to reliably identify eight original auditory stimuli conditions for single trials, and to obtain an assessment of the relevance of any given neuron to the processing of stimulus conditions. Applied to electroencephalography (EEG) data from a visual processing experiment, it was possible not only to confirm results of conventional event-related potential (ERP) analysis, but also to discover a late response component not revealed by ERPs. Applied to fMRI data from an event-related visual object processing experiment,\textsuperscript{15} PyMVPA allowed to identify the original stimulus condition of each trial, and to provide spatio-temporal category specificity profiles without imposing any specific hemodynamic response model.

MVPA methods are in no way limited to processing data from one modality at a time. For example, a reliable description of fMRI data in terms of a simultaneously recorded EEG signal (see Figure 2) allows for identification of areas that are active during any given task, and localization of generators and covariates of dominant EEG frequency bands.\textsuperscript{16} Furthermore, the constructed EEG-to-fMRI mapping can be used for filtering of fMRI and EEG signals, and for EEG-driven interpolation of fMRI timeseries.
To improve the understanding of brain function, neuroscience research requires versatile computing environments and advanced methods that make efficient use of acquired data. Methods developed in the domain of machine and statistical learning are generic, powerful, and their application to neural research has already provided new insights about the brain. Our PyMVPA analysis framework aims to provide a convenient, extensive, and expandable environment to apply existing and to develop new methods for the analysis of neural data. PyMVPA’s user base has been constantly growing and new data analysis methods and methodologies are continuously added to the framework. Future development will further enrich the available techniques and offer promising analysis strategies. One of the immediate next steps will allow for an improved transparent and unbiased model selection. This new functionality will especially help to apply complex non-linear methods while ensuring valid results.17

The authors want to thank all PyMVPA developers and contributors, whose continuous efforts help to forge an open and customizable analysis framework.