0 Task

In this assignment, we will do some simulations in multivariate pattern analysis of fMRI, based on papers [1] and [2]. See appendix for installation and main notes.

1 Pre-processing

At first, you should install all required packages (PyMVPA2, NiBabel, etc...) and locate dataset files in suitable location in your system, We strongly recommend you to use Google/Colab synchronized with github in order to save time while using datasets, packages and task computations.

1.1 Data description

Read first paper and explain details about these variables and concepts of fMRI data on this package:

- Voxel
- Frame
- Maximally active voxels
- Alingment
- Mask
- Correlation on 4D data
- Within/between category correlation

1.2 Load and Visualize data

Load fMRI data of subject 1; plot these images on [25, 40, 100, 125]th frame of time: Set no mask for this part (mask=None in loader function)

- Mid-sagittal cut
- Mid-frontal cut

1.3 Data management

Decompose data into different label-frame pairs; you'll use this mapping for next parts. Example: "Stimuli1": [1, 2, 12, 13, 14] / "Stimuli2": [3, 4, 5, 6, 9, 10] / ...

1.4 Maximally active voxels

With an arbitrary method, find 10 most active voxels; explain your method clearly. For example, one simple method for this section can be computing average of all voxel activities in frame dimension and selecting voxels that are in higher quarter of activity histogram.

[1] Distributed and Overlapping Representations of Faces and Objects in VTC, J.Haxby et al, 2001
[2] A Common, High-Dimensional Model of the Representational Space in Human VTC, J.Haxby, 2011

1.5 Correlation coefficient

The function: $mvpa2.measures.corrcoef.pearson - correlation$ computes pearson correlation of two matrices; implement a method that takes two set of frames and computes their correlation; then upgrade your method to a function that takes two $stimuli - name$ and size "X", then computes their correlation for X frames of each stimuli.

Optional You can use a different correlation method like Kendall or Spearman correlation methods.

1.6 Selective voxels

Add this option to your correlation function; find only non-zero elements of data, name them "masekd"; then find maximally active voxels and name them "maxim"; so your function should be able to do its calculations with excluded or included maxim voxels. note that you should always exclude always zero voxels (these voxels are always zero due to masking)

2 Overlapping and distributed representation of stimuli

Note Use mask $mask4 - vt.nii.gz$ for all of this part.

This is the first main task in this assignment; according to paper "Distributed and Overlapping Representations of Faces and Objects in Ventral Temporal Cortex, J.Haxby et. al" they concluded that object representation in the Ventral Temporal Cortex is distributed and overlapping; so they removed maximall active voxels of a specific object and computed correlations, and classified the data in order to show change in representation details.

2.1 Within category correlations

Using your implemented function, compute correlations for a set of frames of a signle stimuli, and visualize it; that should be a NxN image indicating correlations between each frame with a colormap in this two states:

2.2 Between category correlations

Now, compute overall correlation between set of frames from different categories; that should be a NxN image indivating correlations between each two stimulis like figure no.4 of the paper [1].

- All-voxel data
- Excluded maximally active voxels

2.3 Classification accuracy

To show that these data can be separated by a classifier, one can train a classifier on that and compare accuracies before and after changes; so in this part, train an arbitrary classifier (sklearn.SVM is recommended) in these two states and report accuracy of your classifier.

- All-voxel data
- Excluded maximally active voxels

3 Hyperalignment (+)

In the second paper "A Common, High-Dimensional Model of the Representational Space in Human Ventral Temporal Cortex, J.Haxby et. al" used a new method "hyperalignment" to present a common model not in one subject, but between subject; so then they used PCs to show power of this method in data separation and reported the classification accuracy data.

3.1 Anatomical aligned data classification

Based on library documentation, classifiy between subject data when alignment is anatomical, and save reportes. Your classifier is optional.

3.2 Hyperaligned data classification

Do previous part, with hyper-aligned data; use first 4 subjects as training data and last 2 subjects as test validation data.

A Papers

Distributed and Overlapping Representations of Faces and Objects in Ventral Temporal Cortex[J.Haxby et al, 2001]
A Common, High-Dimensional Model of the Representational Space in Human Ventral Temporal Cortex [J.Haxby et al, 2011]

B Installation

Github/Readme.md

C Library documentation

PyMVPA.org