Spring 1-1-2012

FMRI decoding using sparse neuronal networks

Laura Bernabe Miguel

University of Colorado at Boulder, laura.bernabe.miguel@gmail.com

Follow this and additional works at: https://scholar.colorado.edu/ecen_gradetds

Part of the Electrical and Computer Engineering Commons, and the Neuroscience and Neurobiology Commons

Recommended Citation


https://scholar.colorado.edu/ecen_gradetds/55

This Thesis is brought to you for free and open access by Electrical, Computer & Energy Engineering at CU Scholar. It has been accepted for inclusion in Electrical, Computer & Energy Engineering Graduate Theses & Dissertations by an authorized administrator of CU Scholar. For more information, please contact cuscholaradmin@colorado.edu.
FMRI decoding using sparse neuronal networks

by

Laura Bernabé Miguel

B.A., Escola Politècnica Superior de Castelldefels - UPC, 2005
M.S., Escola Tècnica Superior d’Enginyeria de Telecomunicacions de
Barcelona - UPC , 2010

A thesis submitted to the
Faculty of the Graduate School of the
University of Colorado in partial fulfillment
of the requirements for the degree of
Master of Science Thesis
Department of Electrical, Computer, and Energy Engineering
2012
This thesis entitled:
FMRI decoding using sparse neuronal networks
written by Laura Bernabé Miguel
has been approved for the Department of Electrical, Computer, and Energy Engineering

Prof. François G. Meyer

Prof. Shannon Hughes

Prof. Tor Wager

Date ____________________

The final copy of this thesis has been examined by the signatories, and we find that both the content and the form meet acceptable presentation standards of scholarly work in the above mentioned discipline.
In this thesis we propose the use of Sparse Principal Component Analysis to recover neuronal areas in Brain Imaging. We work with functional magnetic resonance imaging data focusing our attention on the dimensionality reduction stage to represent the neuronal activation within the components that contain the maximum temporal variance, tightly related with the hemodynamic response of the neurons. The motivation for the sparse representation follows the idea of the massive modularity definition of the mind where “different neural circuits are specialized for solving adaptive problems”.

The results show that the new sparse low dimensional basis (Eigenbrains) generated through novel unsupervised algorithms, such as Augmented Sparse Principal Component Analysis, perform competitively in terms of neuronal activity prediction. We push the limits of the brain understanding by describing a neuronal network through each Eigenbrain component and defining a prediction neuronal model using a linear combination of them.
Dedication

To my great family and all my new and old friends.
Acknowledgements

I want to thank the effort and time of all the members that served in the committee for the defense of this Master’s Thesis: Professor Shannon Hughes, Professor Tor Wager and Professor François G. Meyer. I am particularly grateful to my advisor François G. Meyer for working next to me in the development of my research for the last two years.

I really appreciate the important academic contribution of the BBC fellowship program created with the effort of Pete J. Balsells Fundation, Dean Robert Davis and the Generalitat de Catalunya. Education is really basic to continue developing and sharing new knowledge for the whole community and I really believe this is the better way in order to contribute to the growth of Catalonia, specially in this economical difficult period. I am seeking some similar dreams as Mr. Pere Balsells did when he first started his rough path back in the early nineties in USA and I am making the most of this initial support.
# Contents

## Chapter

1 Introduction

1.1 Definition of the problem .................................................. 1

1.2 Pittsburgh Brain Activity Interpretation Competition .................. 3

1.2.1 Functional Brain Image Data ........................................... 4

1.2.2 Rated features ............................................................ 7

1.3 State of the art of fMRI decoding ......................................... 7

1.4 Decoding using sparse representation of brain activity ................. 8

2 Sparse Principal Component Analysis ....................................... 12

2.1 Review of PCA ............................................................... 13

2.1.1 Properties of PCA ....................................................... 15

2.2 Sparsity in PCA ............................................................. 16

2.2.1 Iterative Thresholding Sparse PCA (ITSPCA) ....................... 18

2.2.2 Augmented Sparse PCA (ASPCA) ...................................... 19

2.2.3 Correlation augmented Sparse PCA (CORSPCA) ................. 21

3 Identification of sparse neuronal networks ............................... 23

3.1 Pre-processing ............................................................... 23

3.1.1 Additional pre-processing ............................................... 24

3.1.2 Fitting the input model .................................................. 25
3.2 Neuronal Decoding ................................................................. 26
  3.2.1 Scattering and extension of the neuronal network ($\lambda_1$ and $\lambda_2$) ............... 27
  3.2.2 Further Sparsity ($\lambda_3$) .................................................. 27
3.3 Prediction of brain activity .......................................................... 29
  3.3.1 Our prediction model .......................................................... 29

4 Experiments and discussion ............................................................ 36
  4.1 Design parameters: score vs. sparsity .......................................... 37
  4.2 Eigenbrains ................................................................. 38
  4.3 Prediction performance ........................................................ 40

5 Discussion and conclusion ............................................................ 46

Bibliography ................................................................. 48
### Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Title</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>Acquisition parameters</td>
<td>5</td>
</tr>
<tr>
<td>3.1</td>
<td>Design parameters for maximum prediction score</td>
<td>28</td>
</tr>
<tr>
<td>3.2</td>
<td>$\lambda_3$ values that correspond with Threshold 3</td>
<td>29</td>
</tr>
<tr>
<td>4.1</td>
<td>Design parameters for maximum prediction score</td>
<td>38</td>
</tr>
<tr>
<td>4.2</td>
<td>Sparsity details for maximum prediction score</td>
<td>39</td>
</tr>
<tr>
<td>4.3</td>
<td>Design parameters for optimal neuronal networks architecture</td>
<td>39</td>
</tr>
<tr>
<td>4.4</td>
<td>Sparsity details for optimal neuronal networks architecture</td>
<td>39</td>
</tr>
</tbody>
</table>
## Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.1</td>
<td>3D High-dimensional fMRI data.</td>
<td>2</td>
</tr>
<tr>
<td>1.2</td>
<td>Brodmann areas.</td>
<td>3</td>
</tr>
<tr>
<td>1.3</td>
<td>fMRI experiment scheme.</td>
<td>3</td>
</tr>
<tr>
<td>1.4</td>
<td>Overview of the PBAIC data.</td>
<td>4</td>
</tr>
<tr>
<td>1.5</td>
<td>Rated features from PBAIC data.</td>
<td>8</td>
</tr>
<tr>
<td>1.6</td>
<td>Rated features from PBAIC data.</td>
<td>9</td>
</tr>
<tr>
<td>1.7</td>
<td>Block diagram for fMRI brain decoding</td>
<td>9</td>
</tr>
<tr>
<td>2.1</td>
<td>Example of PCA projections from [1].</td>
<td>14</td>
</tr>
<tr>
<td>2.2</td>
<td>fMRI time series for each voxel from [2].</td>
<td>17</td>
</tr>
<tr>
<td>2.3</td>
<td>ITSPCA algorithm.</td>
<td>19</td>
</tr>
<tr>
<td>2.4</td>
<td>Correlation matrix with higher variance voxels selected.</td>
<td>20</td>
</tr>
<tr>
<td>2.5</td>
<td>Scheme of ASPCA.</td>
<td>20</td>
</tr>
<tr>
<td>3.1</td>
<td>Pipeline for the additional pre-processing.</td>
<td>24</td>
</tr>
<tr>
<td>3.2</td>
<td>(a)(b)(c) Intensity fMRI histograms for Subject 1 for Movie 1, Movie 2 and Movie 3 respectively.</td>
<td>31</td>
</tr>
<tr>
<td>3.3</td>
<td>White-Gray matter for subject1 Slice 16,17,18,19,20 and 21</td>
<td>32</td>
</tr>
<tr>
<td>3.4</td>
<td>Blood Oxygenation Level Dependent contrast fMRI.</td>
<td>32</td>
</tr>
</tbody>
</table>
3.5 (a) Correlation matrix before 1D wavelet across time (b) Correlation matrix after 
1D wavelet across time. .......................................................... 33
3.6 Sorted eigenvalues of the covariance matrix. Label in first 14 (m = 14) ............ 33
3.7 (a) Selection voxels for \(W(\lambda_1)\) and \(W(\lambda_2)\) in \(\Sigma^{\text{spatial}}\). (b) Selection voxels for \(W(\lambda_1)\) 
and \(W(\lambda_2)\) in functional magnetic brain image for \(th_1\) and \(th_2 = 3\) (500 and 150 
voxels respectively). ............................................................... 34
3.8 (a) Tendency of the Variance for the Threshold 1. In red the range of values evaluated 
for \((\lambda_1)\). (b) Tendency of the Variance for the Threshold 2. In red the range of values 
evaluated for \((\lambda_2)\). ................................................................. 34
3.9 Kernel Ridge Regression Prediction score for \(m = 14\) .................................. 35
3.10 (a) Correlation matrix for \(\text{threshold}_3 = 0\) (b) Correlation matrix for \(\text{threshold}_3 = 3\) 35

4.1 Sparsity as a function of \(\text{threshold}_1\), \(\text{threshold}_2\) and \(\text{threshold}_3\). (a)\(\text{threshold}_3 = 
1, \ldots, 11\). (b)\(\text{threshold}_3 = 2, \ldots, 11\). ........................................... 37
4.2 Prediction score as a function of threshold 1(\(\lambda_1\)), thresholds 2(\(\lambda_2\)), and threshold 3(\(\lambda_3\)). 38
4.3 Optimal \(\text{EigenBrains}\) network architecture for Language \(\text{threshold}_1\) and \(\text{threshold}_2 = 
9\) and \(\text{threshold}_3 = 6\). (a) EigenBrain 2 - slices 13/14/15 (b) EigenBrain 4 - slices 
8/9/10 (c) EigenBrain 5 - slices 7/8/9. Slices are numbered from the top to the bottom 40
4.4 Overlap of \(e_2\) (yellow), \(e_3\) (pink), \(e_4\) (green) and \(e_5\) (red) for subspace \(m = 14\) . . . . 41
4.5 Optimal \(\text{EigenBrains}\) network architecture for Faces \(\text{threshold}_1\) and \(\text{threshold}_2 = 
3\) and \(\text{threshold}_3 = 2\). (a) EigenBrain 1 (b) EigenBrain 2 (c) EigenBrain 3 (d) 
EigenBrain 4 ................................................................. 42
4.6 Optimal \(\text{EigenBrains}\) network architecture for Faces \(\text{threshold}_1\) and \(\text{threshold}_2 = 3 
and \(\text{threshold}_3 = 2\). (e) EigenBrain 5 (f) EigenBrain 6 .............................. 43
4.7 Kernel Ridge Regression Prediction score for \(m = 14\) ................................. 43
4.8 Prediction scores vs Sparsity for subspace \(E_{\text{brain}}\) and \(E_{\text{brain}}^1\). (a) Body Parts (b) 
Faces ................................................................. 44
4.9 Prediction scores vs Sparsity for subspaces $E_{brain}$ and $E_{brain}^{3}$. (c) Language (d) Motion
1.1 Definition of the problem

Magnetic resonance imaging (MRI) is a powerful medical sensor technique that constructs a structural image of a scanned area of the body. An MRI machine uses powerful magnetic field to align the magnetization of atomic nuclei in the body, and radio frequency fields to systematically alter the alignment of the magnetization. This causes the nuclei to produce a rotating magnetic moment detectable by the scanner. Functional magnetic resonance imaging (fMRI) is a special type of MRI that is used to detect areas of increased brain activity. When a region of the brain becomes active, blood vessels dilate, allowing freshly oxygenated blood to flow quickly to the region. Oxygen-rich blood disturbs the magnetic field of the scanner less than oxygen-depleted blood. This small change is detectable by the scanner and is used to produce the high-dimensional input data set for a mining process stage.

Functional Magnetic Resonance Imaging is a powerful non-invasive medical technique that generates 3D images and quantifies activation of the brain through the blood-oxygen-level-dependency (BOLD). Intensity variations in the fMRI signal related to the neural activity are very small. Also, they are often overlapped with noise from physiological processes, motion effects or hardware scanner distortions.

Each fMRI sample is represented with some hundred thousands of volume units (voxels) which implies over 120 millions for the whole experiment. Hence, typical brain activity interpretation problems deal with noisy, high-dimensional input datasets.
At the beginning of 20th century, the analysis of fMRI data has been reduced to model the relationship between a simple isolated cognitive stimulus and the 3D image. The main purpose was to identify activated brain regions for a simple and independent task, so the brain was stimulated with a well-known input to isolate the neurological processes. The previous experiments combined with some other brain analysis sensors, such as EEG, ECoG, MEG and PET, helped neuroscience to define masks based on lobar anatomy, cortical and subcortical anatomy, and Brodmann areas.

It is widely known in neuroscience that the cytoarchitectural organization of the human cortex is split into several areas, defined by the German anatomist Korbinian Brodmann. Many of the areas Brodmann defined, based solely on their neuronal organization, have been correlated closely to diverse cortical functions. For example, Brodmann areas 1, 2 and 3 are the primary somatosensory cortex; area 4 is the primary motor cortex; area 17 is the primary visual cortex; and areas 41 and 42 correspond closely to primary auditory cortex. However, functional imaging can only identify the approximate localization of brain activations in terms of Brodmann areas since their actual boundaries in any individual brain requires its histological examination.

Within the last half decade, a new fMRI data analysis challenge has been proposed. The goal is to decode complex neurological functions from natural external stimuli, inferring the relevant voxels for any perceptual, behavioral or emotional input applied to the subject under experimentation. The input mimics a natural environment through software video games, movies, etc. Our research wants to explore to what extent the combination of predictive and interpretable modeling of the neuronal activity can provide new insights into functional brain imaging.
Based on the small-world property of clustered local connectivity of the brain networks [3], our strategy assumes that the low-dimensional basis are well-defined by sparse vectors that are correlated with brain functional modules.

1.2 Pittsburgh Brain Activity Interpretation Competition

The fMRI input data for this study is obtained from the Pittsburgh Brain Activity Interpretation Competition 2006 (PBAIC) as a combination of audio and visual stimulus from recorded videos. The data include the rated set of 30 subjective and objective features that parametrize the brain behavior [4].

The PBAIC was an open competition that involved the analysis of fMRI data of individuals watching three approximately 20-minute segments of movies. It included extensive behavioral ratings of experience coding categories (i.e. human faces, tools, arousal, etc.) and multiple finer
levels (i.e. individual actors, happy or sad emotion, etc.). Conceptually, the challenge was to interpret brain activity sufficiently to be able to predict what an observer was experiencing by looking at fMRI data of their brain. The observer’s experience was quantified in 13 feature ratings, 3 actor presence ratings and 3 location ratings. Accordingly, high quality 3T EPI fMRI data from three subjects viewing three movie clips were provided. For the first two movie clips, 20 minutes of continuous fMRI data and behavioral feature ratings was provided. For the third clip, only functional data was provided. The original goal was to predict the behavioral rating data of each subject. Accuracy of predictions was determined by correlating predicted behavioral ratings with the empirical ratings for each volume acquisition of the fMRI data (1.75s intervals).

![Figure 1.4: Overview of the PBAIC data.](image)

1.2.1 Functional Brain Image Data

Brain image data was collected from three subjects, all of them being native American English speakers.

- Subject 1: male, age 23, right-handed
- Subject 2: female, age 21, right-handed
• Subject 3: male, age 35, right-handed

The fMRI data for the 3 subjects and the 3 movies were captured with the following acquisition parameters:

Table 1.1: Acquisition parameters

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Scanner</td>
<td>Siemens 3T Allegra</td>
</tr>
<tr>
<td>TR/TE</td>
<td>1.75s/25ms</td>
</tr>
<tr>
<td>Flip angle</td>
<td>76 degrees</td>
</tr>
<tr>
<td>Field of view</td>
<td>210 mm</td>
</tr>
<tr>
<td>Slice thickness</td>
<td>3.5 mm</td>
</tr>
<tr>
<td>Slice gap</td>
<td>0</td>
</tr>
<tr>
<td>Number of slices</td>
<td>34</td>
</tr>
<tr>
<td>xy voxel size</td>
<td>3.28125 mm</td>
</tr>
<tr>
<td>xy image dimension</td>
<td>64 × 64</td>
</tr>
<tr>
<td>orientation</td>
<td>axial</td>
</tr>
<tr>
<td>Number of volumes (movie 1)</td>
<td>858 volumes</td>
</tr>
<tr>
<td>Number of volumes (movie 2)</td>
<td>868 volumes</td>
</tr>
<tr>
<td>Number of volumes (movie 3)</td>
<td>900 volumes</td>
</tr>
<tr>
<td>Discarded acquisitions</td>
<td>4 seconds</td>
</tr>
</tbody>
</table>

From the contest, we are given the structural MRI and the functional data. The structural data show the different brain tissue areas and their organization; while the functional data contain the measurement of the blood-oxygen-level-dependency (BOLD) within the activated and non-activated areas.

1.2.1.1 Pre-processed data

The “preprocessed” data set contains the functional and structural data that has been preprocessed with the BrainVoyager analysis software [5]. The data pre-processing attempts to remove some standard artifacts that occur in fMRI experiments which may hinder data analysis. The following pre-processing steps were applied to the functional data:

• **Motion Correction:** This pre-processing option adjusts for small head motion. The first volume of the first movie run was specified as the reference volume to which all others were aligned in space by rigid body transformations. The detected head motion of a volume
with respect to the reference volume results in 3 translation and 3 rotation parameters. These detected values are used to translate and rotate the respective volume accordingly to “undo” the detected head motion.

- **Slice Time Correction:** The slices comprising one functional volume are scanned at different moments in time. For functional analysis, a whole functional volume is treated as one data point, as if all slices were measured at the same time. To make this treatment of the data valid (i.e. for interpreting time the same way across a functional volume), the sequentially scanned slices have to be interpolated in time. This pre-processing step temporally interpolates the slices so that all slices can be treated as if they were acquired at the same time.

- **Linear Trend Removal:** fMRI data measurements are subject to slow, low frequency “drifts” over time, that differ from voxel to voxel. Linear trend removal is accomplished through fitting a line to the time course of each voxel using the least-squares regression method. The obtained oblique line is used to remove the linear trend. Note that the mean is restored, so that the detrended dataset is at the same intensity value as the original data.

### 1.2.1.2 Spatially Normalized data

The spatially normalized data has been pre-processed as per the details in subsection 1.2.1.1, and subsequently spatially normalized. Spatial normalization is a process used to warp the shape of each subject’s brain image into a “standard” brain image space. The purpose is to remove morphological differences between subjects, so that data from different subjects can be directly compared, and so that subject data can be compared to standard brain atlases. Once data has been spatially normalized, the corresponding voxels from subjects’ brain image data matrices will roughly correspond to anatomical brain regions.
1.2.2 Rated features

Rated features are divided into three categories: Base features, Actors, and Locations. See Fig. 1.5 and Fig. 1.6

- **Base features:** Amusement, Attention, Arousal, Body Parts, Environmental Sounds, Faces, Food, Language, Laughter, Motion, Music, Sadness, and Tools
- **Actor features:** Other People, Mark, Randy, Brad, Tim, Jill, Al, and Wilson
- **Location features:** Other Settings, Backyard, Garage, Kitchen, LivingRoom/DiningRoom, and ToolTime
- Other quantitative features (sound amplitude, video brightness, and blank periods) are provided.

1.3 State of the art of fMRI decoding

Some of the standard techniques in the fMRI literature has included in the same process the feature prediction and coordinates selections stages [6, 7, 8, 9] and stochastic statistical techniques are applied to the noisy high-dimensional fMRI data. Other approaches, like our approach, proceed in two stages; implementing the dimensionality reduction techniques within the modeling stage [10, 11, 12]. These are called non-embedded approaches. In the dimensionality reduction stage, we could enforce interpretation of the data. See Fig. 1.7

In the literature, there exists linear and non-linear dimensionality reduction techniques: PCA, kernel PCA, Isomap, locally linear embedding (LLE), Laplacian eigenmaps, etc. All of the low-dimensional subspaces form the previous algorithms are difficult to interpret. It is almost impossible to infer useful knowledge from them because they have lost spatial properties (voxels in our case) and they are represented by too many coordinates.

The use of Sparse PCA low-dimension representations allows us to reduce the number of coordinates, imposing sparsity in the *Eigenbrains* that represent the principal components. Some
Table 1.5: Rated features from PBAIC data.

<table>
<thead>
<tr>
<th>Feature</th>
<th>Description</th>
<th>0</th>
<th>2</th>
<th>4</th>
</tr>
</thead>
<tbody>
<tr>
<td>Faces</td>
<td>Degree to which you see and look at faces on the screen</td>
<td>no face in scene</td>
<td>face and whole body present in scene - face is not the focus of attention</td>
<td>close up of whole face, facial expressions</td>
</tr>
<tr>
<td>Body Parts</td>
<td>Degree to which you see and look at body parts on the screen</td>
<td>nobody in the scene or shots of head/shoulders only</td>
<td>whole body in the scene (e.g., people walking)</td>
<td>focus on one part of the body (e.g., hand shake)</td>
</tr>
<tr>
<td>Tools</td>
<td>Degree to which tools are seen and used on the screen</td>
<td>no tools in scene</td>
<td>closer shot of tool, tool being used (e.g., tool on table)</td>
<td>tool on screen not being used (e.g., tool on shelf)</td>
</tr>
<tr>
<td>Foods</td>
<td>Degree to which food is seen or eaten</td>
<td>no food in scene</td>
<td>closer shot of food not being eaten or prepared (e.g., food sitting on counter)</td>
<td>food being eaten with clear view of what's being eaten</td>
</tr>
<tr>
<td>Motion</td>
<td>Degree to which you see a person or an object moving in the scene; panning is not considered movement</td>
<td>no movement in the scene</td>
<td>NORMAL movement (e.g., walking, cutting vegetables, etc.)</td>
<td>FAST or EXTREME movement (e.g., kids running around the house)</td>
</tr>
<tr>
<td>Language</td>
<td>Degree to which you hear or read language</td>
<td>no talking or written text on screen</td>
<td>normal conversation*</td>
<td>focus on what is being said, just talking with no other sounds, legible written text on screen (e.g., signs, text on slides)</td>
</tr>
<tr>
<td>Environmental Sounds</td>
<td>Degree to which you hear sounds other than language, music and laugh track; Talk Time</td>
<td>no sounds in scene (people can be talking, music can be playing, laugh track can be heard)</td>
<td>sound at normal volume but isn't the focus of attention (e.g., Tim talking in the background while Ali hammers)</td>
<td>sound overwhelms any other sound (e.g., power tool being used)</td>
</tr>
<tr>
<td>Music</td>
<td>Degree to which you hear music in the scene</td>
<td>no music in scene</td>
<td>music at normal volume but not focus of attention</td>
<td>music is the focus of attention, people reading to the music (e.g., dancing, singing), no talking</td>
</tr>
</tbody>
</table>

Figure 1.5: Rated features from PBAIC data.

Functional regions related to Brodmann areas are discovered by the sparse estimators.

1.4 Decoding using sparse representation of brain activity

In this thesis we propose a new method to analyze fMRI data. It is organized in two major steps: (a) the implementation of a cognitive decoding model, which interprets cognitive processes; (b) the development of a brain behavioral predictive model, which evaluates the performance through the rated features.
The backbone for the decoding brain activation stage is the detection of significant brain activity through the maximum temporal variation within the voxels in the brain. The previous context fits in the classical principal component analysis (PCA) dimensionality reduction method; nevertheless, it is a poor subspace estimator for our experiments because we focus on high dimensional observations.
We introduce some notations. Let $n$ be the number of time samples or the number of scans; and $p$ the number of voxels in each scan. Let $x(t) \in \mathbb{R}^p$ be a 3D scan collected at a time $t$, that is reshaped as a row vector.

$$x(t) = \left[ x_1(t) \ldots x_p(t) \right].$$

Overall, the entire fMRI dataset can be described as a matrix;

$$X = \begin{bmatrix} x_1(1) & \cdots & x_p(1) \\ \vdots & \vdots & \vdots \\ x_1(n) & \cdots & x_p(n) \end{bmatrix}.$$

The number of time samples ($n \approx 850$) is significantly smaller than the number of voxels ($p = 64 \times 64 \times 34$), so the spatial covariance matrix of the input data is not a good estimator of the time variances trends.

In addition, we know that $X$ contains spatially localized significant variables, which means that the principal modes of variation should be localized as well. Also motivated by the concept of Massive Brain Modularity (MMH) - which says that the cognitive brain is conformed by sparse neuronal modules which interact adaptively in order to develop a specific task [13] - we use novel Sparse Principal Component Analysis (SPCA) techniques to decode the brain into meaningful sparse neuronal modules, that we called Eigenbrains ($e_j$).

According to the MMH, the mind consists of a multitude of domain-specific modules or mental organs, each of them with a specialized design that makes it an expert in one area of interaction with the world. Being domain specific, each module is activated by, and only by, mental representations of the problem(s) in its area of expertise [14].

The most significant Sparse Principal Components (the first $m$ with higher variance) are carefully interpreted as active neuronal modules. They sketch some interesting Brodmann areas whose functions are related to the interaction with a movie: primary and auditory association cortex; primary, secondary and associative visual cortex ($V1, V2, V3$); prefrontal cortex and frontal eye fields, etc.
We organize the sparse principal components in the columns of the $E_{\text{brain}}$ rotation matrix $\in \mathbb{R}^{p \times m}$, creating the low-dimensional quasi orthogonal basis. The high-dimensional data is projected onto $E_{\text{brain}}$ to create $X^{nb}$:

$$X^{nb} = X \cdot E$$

(1.1)

And $X^{nb}$, the low-dimensional data, is defined as follows;

$$X^{nb} = \begin{bmatrix}
  x_{1}^{nb}(1) & \cdots & x_{m}^{nb}(1) \\
  \vdots & \ddots & \vdots \\
  x_{1}^{nb}(n) & \cdots & x_{m}^{nb}(n)
\end{bmatrix}$$

Such functional networks described by the neurological subspace ($E_{\text{brain}}$) are allowed to be different for each feature, hence, different sets of relevant voxels are selected. The structural design for each network is controlled by the scattering ($\lambda_1$ and $\lambda_2$) and the sparsity ($\lambda_3$) parameters.

Therefore, an optimal sparse neuronal set of Eigenbrains exist for each feature ($f$). We linearly combine them in order to implement the supervised ridge regression prediction model;

$$f(t) = x_{1}^{nb}(t)\beta_1 + \ldots + x_{m}^{nb}(t)\beta_m$$

(1.2)

for $t = 1, \ldots, n$. The previous expression shows how much activity is represented through each Eigenbrain; and how the domain-specific modules of knowledge and psychological structures are correlated with the output rated features.
Chapter 2

Sparse Principal Component Analysis

In many contemporary datasets, such as in the case of our fMRI study, if we organize the $p$-dimensional observations to be the rows of an $n \times p$ data matrix $X$; the number of variables $p$ is often comparable to, or even much larger than the sample size $n$. For example,

- **Image recognition**: The face recognition problem typically has $p = 1.6 \times 10^6$ observations and the database may contain only a few hundred pictures.

- **Shape Analysis**: There is a class of methods for analyzing the shape of an object based on repeated measurements that involves annotating the objects for landmarks. The landmarks act as dimensions of the objects.

- **Chemometrics**: In many chemometric studies the data consists of several thousand spectra measured at several hundred wavelength positions.

- **Climate studies**: Measurements on atmospheric indicators are taken at a number of monitoring locations over a large temporal extend.

- **Functional data Analysis**: A speech dataset example consists of 162 observations, each of which is a periodogram of a phoneme spoken by a person. Our fMRI experiment focuses on over 850 samples (for each video and subject), and each one of them has $p \approx 10^5$ dimensions.
• Microarray analysis: Gene microarrays present data in the form of expression profiles of several thousand genes for each subject under study.

One of the crucial issues in the analysis of large $p$ datasets is the dimensionality reduction of the feature space[15].

2.1 Review of PCA

Principal component analysis is a variable reduction procedure. This technique is useful when you have obtained data on a number of variables (possibly a large number of variables), and believe that there is some redundancy in those variables. In this case, redundancy means that some of the variables are correlated with one another, possibly because they are measuring the same construct. If there is redundancy, it should be possible to reduce the observed variables into a smaller number of principal components (artificial variables), which account for most of the variance in the observed variables. Technically, a principal component can be defined as a linear combination of optimally-weighted observed variables.

Hence, the central idea of PCA is to reduce the dimensionality of a data set in which there are a large number of interrelated variables, while retaining as much of the variation present in the data set as is possible. This reduction is achieved by transforming into a new set of variables, the principal components, which are uncorrelated, and which are ordered so that the first few retain most of the variation present in the original set of variables.

This unsupervised method starts with $n$ $p$-dimensional observation vectors, which can be summarized by projecting down onto a $m$-dimensional subspace. The summary is the projection of the original vectors onto the $m$ directions of the principal components subspace.

$$E_{\text{brain}} = \begin{bmatrix} e_1(1) & \cdots & e_m(1) \\ \vdots & \vdots & \vdots \\ e_1(p) & \cdots & e_m(p) \end{bmatrix}$$

There are several equivalent ways of deriving the principal components mathematically. The
The simplest one is by finding the projections which maximize the variance. The first principal component is the direction in feature space along which projections have the largest variance. The second principal component is the direction which maximizes variance among all directions orthogonal to the first. The $k_{th}$ component is the variance-maximizing direction orthogonal to the previous $k − 1$ components. There are a total of $p$ principal components.

The goal is to project the p-dimensional feature vectors onto a line through the origin ($e \in \mathbb{R}^p$), so the residual projected error is minimized:

$$\min \| x_i - \langle x_i, e \rangle \cdot e \|^2 = \min \| x_i \|^2 - 2 \langle x_i, e \rangle^2 + 1 \quad (2.1)$$

As the first term does not depend on $e$, it is equivalent to the following expression for all $i$:

$$\max \| x_i \|^2 - 2 \langle x_i, e \rangle^2 = \max \text{Var} \langle x_i, e \rangle \quad (2.2)$$

If we stack all the samples in a matrix $X$, then the previous optimization problem can be expressed through its covariance matrix $\Sigma_{ij}$ for all $i$ and $j$.

$$\Sigma_{ij} = \text{cov}(x_i, x_j) = E \left[ (x_i - \mu_i) \cdot (x_j - \mu_j)^T \right] \quad (2.3)$$

$$\max \ e^T \frac{X^T X}{n} e = \max \ e^T \Sigma e \quad (2.4)$$
Finally, by using the Langrage Multipliers variables and its derivatives, we infer that the principal components correspond with the eigenvectors of the covariance matrix of the input data. Therefore, computation of the principal components reduces to the solution to an eigenvalue-eigenvector problem for a positive-semidefinite symmetric matrix.

\[ \Sigma e = \lambda e \] (2.5)

### 2.1.1 Properties of PCA

This is a good place to remark that if the data really fall in m-dimensional subspace, created by the first m principal components, then \( \Sigma \) will have only m positive eigenvalues; because, after subtracting off those components, there will be no residuals. The other \( p - m \) eigenvectors will all have eigenvalue 0. If the data cluster around a m-dimensional subspace, then \( p - m \) of the eigenvalues will be very small.

\[ R^2 = \frac{\sum_{i=1}^{m} l_i}{\sum_{j=1}^{p} l_j} \] (2.6)

where \( l_i \) and \( l_j \) are eigenvalues, and \( R^2 \) is the fraction of the original variance of the dependent variable retained by the fitted values. Projections onto the first two or three principal components can be visualized; however, they may not be enough to really give a good summary of the data. Usually, to get an \( R^2 \) of 1, you need to use all p principal components. How many principal components you should use depends on the data. In some fields, you can get better than 80% of the variance described with just two or three components.

We have not assumed that the data are drawn at random from some distribution, nor have we assumed that the different rows of the data frame are statistically independent. This is because no such assumption is required for principal components. We simply say these data can be summarized using projections along these directions but nothing about the larger population or stochastic process the data came from.
However, we could add a statistical assumption and see how PCA behaves under those conditions. The simplest one is to suppose that the data are i.i.d draws from a distribution with covariance matrix $\Sigma$. Then the sample covariance matrix will converge on $\Sigma_0$ as $n \to \infty$. Since the principal components are smooth functions of $\Sigma$ (eigenvectors), they will tend to converge as $n$ grows. So, along with that additional assumption about the data-generating process, PCA does make a prediction: in the future, the principal components will look like they do now.

The regular principal components are difficult to interpret because they are a linear combination of most of the components in the high-dimensional space. The goal is to represent the fMRI data into some other basis that is more compact and abstract and, indeed, easier to interpret and generate more accurate future predictive models. It is really difficult to infer neurological processes within the $10^5$ voxels.

Indeed, as $n << p$ the covariance matrix is a very poor estimator for the temporal trends so we need to seek a sub-covariance matrix within the original $\Sigma \in \mathbb{R}^{n \times p}$. In order to do that, we can use the natural structure of the fMRI that contains spatially localized significant voxels, and try to separate them from the rest of the voxels.

### 2.2 Sparsity in PCA

In the approach used in *Neuronal Decoding for fMRI images* we focus on measuring brain activation that is related with the temporal variation of all the voxels across time. See Fig. 2.2 We define the $\Sigma^{spatial}$ covariance matrix across the voxels as the original input for the SVD decomposition.

\[
\Sigma_{ij} = \text{cov}(x_i, x_j) = E \left[ (x_i - \mu_i) \cdot (x_i - \mu_i)^T \right] \tag{2.7}
\]

where $i, j = 1, \ldots, p$ are all the voxels in our brain.

Therefore, given the covariance matrix in the space domain $\Sigma^{spatial}$, Sparse PCA can be cast as a cardinality-constrained quadratic program, maximizing the variance with a sparse vector $e_j$. 

Figure 2.2: fMRI time series for each voxel from [2].

having no more than $k$ non-zero elements.

$$
\max \quad (e^T \Sigma_{spatial} e)
$$

subject  \quad e^T e = 1

$$
\|e_j\|_0 \leq k
$$

(2.8)

This optimization problem is non-convex, NP-hard and therefore intractable [16]. Therefore, it has entailed the development of novel mathematic algorithms in the unsupervised dimensionality reduction literature within the half decade such as: Iterative thresholding sparse PCA (ITSPCA) by Ma (2011) [17], Augmented sparse PCA (ASPCA) by Paul and Johnstone (2007) [15], Correlation augmented sparse PCA (CORSPCA) by Nadler (2009)[18], Sparse PCA via regularized SVD (sPCA-rSVD) by Shen and Huang(2008)[19], Generalized Power Method for Sparse Principal Component Analysis by Jorne,Nesterov,Richtrik and Sepulchre (2010) [20] , etc.

Different thresholding techniques are used to reduce the number of dimensions from the regular principal components; but they do not ensure complete orthogonality among the components within the low-dimensional subspace, although we will see in the following sections that they are pretty close to be uncorrelated.

Augmented Sparse PCA (ASPCA) is considered the best tool for our experiments because we
can make a biological interpretation throughout all the stages of the algorithm. We can understand the low dimension *Eigenbrains* subspace in terms of neurological processes.

Most developments in sparse PCA methodologies typically start with a certain optimization formulation of PCA and then induce a sparse solution by introducing appropriate penalties or constraints. Moreover, when Σ has sparse leading eigenvectors it becomes possible to estimate them consistently under high-dimensional settings.

It is also necessary to estimate the background noise variance for the experiments for all the approaches in order to normalize the data. Assuming normality distributions of the observations we calculate its value as $\hat{\sigma}^2 = \text{median}(\text{Var}(x_j))$.

Some of the novel algorithms to estimate the sparse PCA components are reviewed in the following section.

### 2.2.1 Iterative Thresholding Sparse PCA (ITSPCA)

The ITSPCA paper [17] by Ma focus on finding principal subspaces of $S$ (variable used to defined the covariance matrix in the previous paper) spanned by sparse leading eigenvectors, as opposed to finding each sparse vector individually. One of the reasons for this is that individual eigenvectors are not identifiable when some of the leading eigenvalues are identical or close to each other. Also, if we view PCA as a dimension reduction technique, it is the low-dimensional subspace onto which we project data that is of the greatest interest. A new iterative thresholding algorithm is proposed to estimate principal subspaces, which is motivated by the orthogonal iteration method in the matrix computation literature. In addition to the usual steps of orthogonal iteration, an additional thresholding step is added to seek a sparse basis for the subspace.

When the covariance matrix follows the spiked covariance model [21], the algorithm is shown to yield a uniformly consistent subspace estimator.
2.2.2 Augmented Sparse PCA (ASPCA)

For the more general multiple component case, Paul and Johnstone [15] proposed an augmented sparse PCA method to estimate each of the leading eigenvectors. They showed that their procedure attains near optimal rate of convergence under a range of high-dimensional sparse settings when the leading eigenvalues are comparable and well separated.

The motivation for this approach considers the SPCA estimation scheme studied by Johnstone and Lu (2004)[22]. Suppose you have calculated the sample variances of all the coordinates; i.e, diagonal terms of Σ (being the covariance matrix) are denoted by σ₁²,...,σₚ². The general pipeline of the method is as follows:

- Define I to be the set of indices k ∈ {1,...,p} such that σₖ² > λ
- Let Σᵢ,I be the submatrix of Σ corresponding to the coordinates I. Perform an eigen analysis of Σᵢ,I. Denote the eigenvectors by e₁,...,eₙ (n = number of observations).
• For $v = 1, \ldots, m$ ($m = \text{number of the principal components in the subspace}$), estimate $\mu_v$ by augmenting zeros to all the coordinates that are in $\{1, \ldots, p\}$.

![Figure 2.4: Correlation matrix with higher variance voxels selected.](image)

Johnstone and Lu showed that if one chooses an appropriate threshold $\lambda$ then the estimator is consistent under the sparsity constraint.

![Figure 2.5: Scheme of ASPCA.](image)

The ASPCA aims to address the problem of estimating eigenvectors from a minimax risk analysis viewpoint. Henceforth, the observations are assumed to have a Gaussian distribution. With this condition, the Augmented Sparse Principal Component Analysis (ASPCA) has an optimal rate of convergence over suitable regularity conditions. Moreover, it is also assumed that the leading eigenvalues of the population covariance matrix are distinct, so the eigenvectors are identifiable.
We will go into more details about the ASPCA algorithm and how we adapt it to the Neurological Decoding approach in Chapter 3.

2.2.3 Correlation augmented Sparse PCA (CORSPCA)

The CORSPCA analysis in [18] predicted that there would be some gap between the performance of the sparse PCA from Johnstone and Lu (2004) and the possible optimal one. Nadler discussions refers to the work of Bicke and Levina (2008) [23], which assumes that the covariance matrix is sparse, but not necessarily that the eigenvectors are sparse. The key observation is that the assumption of Johnstone and Lu - that individual signals are simultaneously sparse in some unknown basis - implies more than just having relatively few features with large variance. It also implies that these features should be highly correlated among themselves.

Under the assumption of uncorrelated Gaussian noise, this observation suggests an alternate feature selection approach. Rather than working only with the covariance matrix, the structure of the correlation matrix is also analyzed to look for highly correlated variables.

The suggested procedure is as follows:

- Given a data matrix $X$, compute the covariance ($\Sigma$) and correlation matrices ($C$).

- Estimate the noise variance $\sigma^2$.

- Find the sure signal features:

$$I_s = \left\{ i, \frac{\Sigma(i,i)}{\sigma^2} > 1 + \sqrt{\frac{2}{n}} t(\alpha, p) \right\}$$

where

$$t(\alpha, p) = \sqrt{2 \ln p} - \frac{\ln(4\pi \ln p)}{2\sqrt{2 \ln p}} - \frac{\ln \alpha}{\sqrt{2 \ln p}}$$

and $\alpha$ is the confidence level chosen by the user.

- For each $i = 1, \ldots, p$ and $i \notin I_s$ compute

$$E_i = \frac{1}{|I_s|} \sum_{j \in I_s} C(i, j)^2$$
• Denote by $I_c$ the set of variables highly correlated to those in $I_s$: 

$$I_c = \left\{ j, j \notin I_s, E_j > \frac{1}{n-1} (1 + \sqrt{2t(\alpha,p)}) \right\}$$
Chapter 3

Identification of sparse neuronal networks

The pipeline for our approach is described by three major steps: (i) Pre-processing, (ii) Neuronal Decoding and (iii) Prediction of brain activity.

3.1 Pre-processing

In general, fMRI data is really rich and may be grouped into signals of interest and signals not of interest. The signals of interest are task-related and function-related. The task-related signals are the easiest to model. They are the responses of the brain to a given task. It is conceivable that there are several different types of transiently task-related signals coming from different regions of the brain. The function-related signals manifest as similarities between voxels within a particular functional domain. The signals not of interest include physiology-related, motion related, and scanner-related signals. Physiology-related signals such as breathing and heart rate tend to come from the brain ventricles (fluid-filled regions of the brain) and areas with large blood vessels present, respectively. Motion-related signals can also be present, and tend to show up as changes across large regions of the image (particularly at the edges of images) [24].

Additionally, changes in the fMRI signal that occur during brain activation are small (1–5%) and are often contaminated by noise (created by the imaging system hardware or physiological processes), so the cleaner the more sensitive will be the brain activity detection.

In all of our experiments, the input matrix $X$ is created with the pre-processed and spatially normalized fMRI data from the PBAIC contest (See subsections 1.2.1.1 and 1.2.1.2). We do an
additional pre-processing step prior to this approach to clean the data. See Fig. 3.1

\begin{align*}
X & \in \mathbb{R}^{n \times p} \\
\downarrow & \\
\text{DOWNSAMPLE} & \\
\downarrow & \\
\text{BRAIN ISOLATION} & \\
\downarrow & \\
\text{CLEAN GLOBAL TRENDS} & \\
\end{align*}

Figure 3.1: Pipeline for the additional pre-processing.

Finally, the data is fitted into the assumed input models for the sparse PCA estimators: \textbf{independence} between samples and \textbf{spiked population model}.

3.1.1 Additional pre-processing

The \( n \) fMRI data samples for our experiments are stacked in \( X \in \mathbb{R}^{n \times p} \) where each row is filled with the \( p \) voxels from a 3D brain image. \( X \) is down-sampled and the white-gray matter from all the samples is separated from the outer head area, skull and intra-cranial CSF to build \( X_{gw} \). In this way, high intensity signals from physiological effects are removed from neuronal signals. In order to do that we use standard histogram thresholding techniques with the fMRI signal intensity. See Fig 3.2 and Fig 3.3.

Finally, we subtract the first and the second principal component of the time covariance matrix \( \Sigma_{time} \in \mathbb{R}^{n \times n} \) from \( X_{gw} \) in order to remove global trend effects - i.e., related to motion artifacts from the head of subject.

\begin{align*}
\Sigma_{ij}^{time} &= \text{cov}(x_i, x_j) = E \left[ (x_i - \mu_i) \cdot (x_j - \mu_j)^T \right] \\
X_{gw} &= X_{gw} - \langle X_{gw}, \text{pca}_{time} \rangle \text{pca}_{time} \\
\end{align*}

(3.1)
3.1.2 Fitting the input model

In order to fit the observations into the input data model, they need to be independent and the background noise needs to be white. Each observation is created by the addition of the signal of interest and the signal not of interest;

\[ x(t) = u_t e + \sigma z_t, \]  

(3.2)

where \( e \in \mathbb{R}^p \) is the component to be estimated, \( u_t \sim N(0,1) \) are i.i.d. Gaussian random effects, \( \sigma \) is the p-vector representation of the background noise level, and \( z_t \sim N_p(0, I) \) are independent p-dimensional noise vectors [22].

It is broadly known that fMRI images are related through the hemodynamic response pattern, so, they are not independent.

It has been revealed in some studies that the temporal variation of the BOLD activation is essentially Gaussian [25], so it suffices to uncorrelate each observation to make them independent.

Wavelets have been widely used in various signal and image processing contexts since their mathematical development in the late 1980s, including many prior applications to image compression, non-parametric regression, and problems in brain mapping. However, the single most important property of the discrete wavelet transform is that the correlation between the wavelet coefficients of a signal will generally be small even if the signal itself is highly autocorrelated in time. This is sometimes called the whitening or decorrelating property of the wavelet transform [26]. Hence, by applying 1D wavelet transform to each voxel time sequence values \( x_j \) for \( j = 1, \ldots, p \) we decorrelate the \( n \) observations of the input sets (see Fig 3.5). We recall the definition of the correlation matrix as:

\[ \text{coor}(x, y) = \frac{E[(x-\mu_x)(y-\mu_y)]}{\sigma_x \sigma_y}. \]

Note that the time scale is no longer in seconds.

The spatial covariance matrix \( \Sigma^{\text{spatial}} \) must fit in the spiked population model to be an optimal input set for ASPCA. This implies that the higher \( m \) eigenvalues \( l_i \) for \( i = 1, \ldots, m \) are above the background noise(\( \sigma^2 \)). See Fig 3.6. We apply 3D wavelet transform to potentiate the
intrinsic sparsity of the fMRI images in the new basis. Hence, the entries in $x(t)$ are wavelet coefficients from now on.

As $\sigma^2$ is unknown for the fMRI input dataset and assuming normal distributions for the observations, we estimated its value as $\hat{\sigma^2} = \text{median}(\text{Var}(x_j(t)))$

### 3.2 Neuronal Decoding

The most significant Sparse Principal Components obtained through ASPCA are interpreted as active neuronal modules. They sketch some interesting Brodmann areas, whose functions are related to the audio and visual frames of a movie: primary and auditory association cortex, primary and secondary visual cortex $V1$ and $dV2$, associative visual cortex $V3$, prefrontal cortex and frontal eye fields, etc.

We use the first $m = 14$ *Eigenbrains* as the new basis for the low dimensional subspace. As mentioned earlier, the core of the sparse PCA idea focuses on how to choose the appropriate coordinates and all the observations are normalized with the level of background noise $\sigma^2$.

Motivated by ASPCA by Paul and Johnstone [15] we apply the algorithm as follows:

1. Threshold 1: Extract the voxels with time variance exceeding $\lambda_1$ to build($\mathcal{W}(\lambda_1)$). These are the voxels that correspond with neuronal activity. Compute the normalized factor within the coordinates in the previous set. See Fig 3.8

2. Threshold 2: Compute the normalized correlation between the voxels from $\mathcal{W}(\lambda_1)$ and $\mathcal{W}(\lambda_1)^C$; and select the higher coordinates (exceeding $\lambda_2$) to build($\mathcal{W}(\lambda_2)$). See Fig 3.8. It defines the extension of the neuronal network.

3. Take the union of $\mathcal{W}(\lambda_1) \cup \mathcal{W}(\lambda_2)$. The cardinality of the previous set tunes the cluster properties of the brain modules.

4. Compute PCA on $(x_k|k \in \mathcal{W}(\lambda_1) \cup \mathcal{W}(\lambda_2))$ and pad them with zeros to extend the coordinates creating $e_i \in \mathbb{R}^p$. 
(5) Threshold 3: Hard threshold the coordinates in \( e_i \) according to the convergence decay factor \( (\lambda_3) \) for regular PCA, in order to filter out the noise.

\[
\lambda_i^3 = \sqrt{\frac{\log(b)(1+l_i)}{nl_i^2}}
\]

(3.3)

where \( i = 1, \ldots, m \) and \( b = \text{card}(W(\lambda_1) \cup W(\lambda_2)) \)

(6) Normalize \( e_i \) to obtain the eigenbrains.

Finally, we project the high-dimensional input data into \( E_{brain} \) creating \( X^{nb} \in \mathbb{R}^{n \times m} \) where each row corresponds to an observation in the low-dimensional space \( (x^{nb}(t) \in \mathbb{R}^m) \).

3.2.1 Scattering and extension of the neuronal network (\( \lambda_1 \) and \( \lambda_2 \))

The threshold 1 and threshold 2, defining the cut-off parameters \( \lambda_1 \) and \( \lambda_2 \), are related to the extension of the original neuronal network that compose the Eigenbrain. \( \lambda_1 \) selects the voxels with higher variance related to the brain activation, and \( \lambda_2 \) selects the voxels with smaller variance which are strongly related to the previous activated voxels. The values of the first two thresholds are related because they both scatter the voxels of the network.

To be able to analyze the most significant and interesting combinations of these two thresholds we create a simplified system through labels that represent significant and meaningful values according to the tendency, shown in Fig. 3.7. For threshold 1 and threshold 2 we create 10 labels for 20 different values of \( \lambda_1 \) and \( \lambda_2 \) (corresponding to \( \gamma_{1,n} \) and \( \gamma_{2,n} \) in the original ASPCA algorithm). They have been chosen according with our own simulated experimental results in MATLAB. The higher the values for threshold 1 - threshold 2, the more voxels are selected for the sets \( W(\lambda_1) \) and \( W(\lambda_2) \) respectively.

3.2.2 Further Sparsity (\( \lambda_3 \))

The third threshold is used to filter out the noise and shrink the smallest entries for the regular PCA applied over the union of \( W(\lambda_1) \cup W(\lambda_2) \). The parameter \( \lambda_3 \) tunes the uncorrelation
Table 3.1: Design parameters for maximum prediction score

<table>
<thead>
<tr>
<th>Threshold 1 - Threshold 2</th>
<th>λ₁</th>
<th>λ₂</th>
<th>card((W(λ₁)))</th>
<th>card((W(λ₂)))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>326.2</td>
<td>47.6</td>
<td>300</td>
<td>90</td>
</tr>
<tr>
<td>2</td>
<td>268.7</td>
<td>38.2</td>
<td>400</td>
<td>120</td>
</tr>
<tr>
<td>3</td>
<td>233.0</td>
<td>32.4</td>
<td>500</td>
<td>150</td>
</tr>
<tr>
<td>4</td>
<td>202.9</td>
<td>25.8</td>
<td>600</td>
<td>180</td>
</tr>
<tr>
<td>5</td>
<td>174.3</td>
<td>21.6</td>
<td>700</td>
<td>210</td>
</tr>
<tr>
<td>6</td>
<td>150.7</td>
<td>19.0</td>
<td>800</td>
<td>240</td>
</tr>
<tr>
<td>7</td>
<td>133.0</td>
<td>16.8</td>
<td>900</td>
<td>270</td>
</tr>
<tr>
<td>8</td>
<td>118.2</td>
<td>15.1</td>
<td>1000</td>
<td>300</td>
</tr>
<tr>
<td>9</td>
<td>105.9</td>
<td>13.6</td>
<td>1100</td>
<td>330</td>
</tr>
<tr>
<td>10</td>
<td>96.4</td>
<td>12.2</td>
<td>1200</td>
<td>360</td>
</tr>
</tbody>
</table>

and the further sparsity of Eigenbrains in the new basis. This parameter hard thresholds the entries \(i = 1, \ldots, p\) for the different Eigenbrains \((e_j(i))\). See Equation 3.4

\[
e_i(j) = \begin{cases} 
  e_i(j), & |e_i(j)| > \lambda_3^i, \\
  0, & |e_i(j)| \leq \lambda_3^i,
\end{cases}
\]  

(3.4)

where \(i = 1, \ldots, m\) and \(j = 1, \ldots, p\).

The hard thresholding step in threshold 3 has been organized into 11 labels that are closely related to the final sparsity and correlation of the principal components. The parameter \(\lambda_3\) which shrinks the smallest values in each Eigenbrain \((e_j\) for \(j = 1, \ldots, m\) is more aggressive the bigger the \(j\) index is, so it is not a constant threshold for each label (see Table 3.2 and Fig 3.9). The increase of the \((\lambda_3^j)\) is related to the convergence decay factor for the regular PCA (see equation 3.3). It corresponds to \(\gamma_{3,n}\) in the original ASPCA algorithm.

If \(\lambda_3\) is small, the Eigenbrains are more orthogonal; so, this parameter determines the correlation of the components within the low-dimensional subspace. Fig 3.10
Table 3.2: \( \lambda_3 \) values that correspond with Threshold 3

<table>
<thead>
<tr>
<th>Threshold 3</th>
<th>((\lambda_1^3))</th>
<th>((\lambda_2^3))</th>
<th>((\lambda_3^3))</th>
<th>((\lambda_4^3))</th>
<th>((\lambda_5^3))</th>
<th>((\lambda_6^3))</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>0</td>
</tr>
<tr>
<td>2</td>
<td>0.0088</td>
<td>0.0130</td>
<td>0.0155</td>
<td>0.0159</td>
<td>0.0179</td>
<td>0.0190</td>
</tr>
<tr>
<td>3</td>
<td>0.0124</td>
<td>0.0184</td>
<td>0.0219</td>
<td>0.0226</td>
<td>0.0254</td>
<td>0.0269</td>
</tr>
<tr>
<td>4</td>
<td>0.0152</td>
<td>0.0225</td>
<td>0.0268</td>
<td>0.0276</td>
<td>0.0311</td>
<td>0.0330</td>
</tr>
<tr>
<td>5</td>
<td>0.0176</td>
<td>0.0260</td>
<td>0.0310</td>
<td>0.0319</td>
<td>0.0359</td>
<td>0.0381</td>
</tr>
<tr>
<td>6</td>
<td>0.0196</td>
<td>0.0290</td>
<td>0.0346</td>
<td>0.0357</td>
<td>0.0401</td>
<td>0.0426</td>
</tr>
<tr>
<td>7</td>
<td>0.0215</td>
<td>0.0318</td>
<td>0.0379</td>
<td>0.0391</td>
<td>0.0440</td>
<td>0.0504</td>
</tr>
<tr>
<td>8</td>
<td>0.0232</td>
<td>0.0344</td>
<td>0.0410</td>
<td>0.0422</td>
<td>0.0475</td>
<td>0.0444</td>
</tr>
<tr>
<td>9</td>
<td>0.0249</td>
<td>0.0367</td>
<td>0.0438</td>
<td>0.0451</td>
<td>0.0508</td>
<td>0.0467</td>
</tr>
<tr>
<td>10</td>
<td>0.0264</td>
<td>0.0390</td>
<td>0.0465</td>
<td>0.0478</td>
<td>0.0538</td>
<td>0.0539</td>
</tr>
<tr>
<td>11</td>
<td>0.0278</td>
<td>0.0411</td>
<td>0.0490</td>
<td>0.0504</td>
<td>0.0567</td>
<td>0.0571</td>
</tr>
</tbody>
</table>

3.3 Prediction of brain activity

Possibly the most elementary algorithm that can be kernelized is ridge regression. Ridge Regression is used to find a linear function that models the dependencies between covariates \( \{x_i\} \) and response variables \( \{f_i\} \), both continuous. The classical way to do this is to minimize the quadratic cost:

\[
\min \frac{1}{2} \sum_i (f_i - \beta^T x_i)^2.
\]  

(3.5)

To avoid overfit, it is necessary to regularize which can be done by simply penalizing the norm of \( \beta \). This is sometimes called weight-decay and it is chosen according to a cross validation process.

\[
\min \frac{1}{2} \sum_i (f_i - \beta^T x_i)^2 + \frac{1}{2} \epsilon \|\beta\|^2.
\]  

(3.6)

By introducing Lagrange multipliers into the problem the derivation becomes similar to that of the support vector machine problem.

3.3.1 Our prediction model

The prediction of a feature \( \hat{f} \) at an unknown time \( t_x \) is implemented as a supervised learning classifier. We formulate the prediction model using Kernel Ridge Regression;
\[
\hat{f}(t_x) = \sum_{i=1}^{n} \hat{\alpha}_i \cdot \kappa(x_{nb}^t, x_{nb}^{tx}), \tag{3.7}
\]

where \( t = 1, \ldots, n \) represent the training set, \( \hat{\alpha} \) is weighted by the previous set, and \( \kappa \) is a Gaussian kernel.
Figure 3.2: (a)(b)(c) Intensity fMRI histograms for Subject 1 for Movie 1, Movie 2 and Movie 3 respectively.
Figure 3.3: White-Gray matter for subject1 Slice 16,17,18,19,20 and 21.

Figure 3.4: Blood Oxygenation Level Dependent contrast fMRI.
Figure 3.5: (a) Correlation matrix before 1D wavelet across time (b) Correlation matrix after 1D wavelet across time.

Figure 3.6: Sorted eigenvalues of the covariance matrix. Label in first 14 (m = 14)
Figure 3.7: (a) Selection voxels for $W(\lambda_1)$ and $W(\lambda_2)$ in $\Sigma_{\text{spatial}}$. (b) Selection voxels for $W(\lambda_1)$ and $W(\lambda_2)$ in functional magnetic brain image for $\text{th}_1$ and $\text{th}_2 = 3$ (500 and 150 voxels respectively).

Figure 3.8: (a) Tendency of the Variance for the Threshold 1. In red the range of values evaluated for $(\lambda_1)$. (b) Tendency of the Variance for the Threshold 2. In red the range of values evaluated for $(\lambda_2)$. 
Figure 3.9: Kernel Ridge Regression Prediction score for $m = 14$

Figure 3.10: (a) Correlation matrix for threshold $3 = 0$ (b) Correlation matrix for threshold $3 = 3$
Chapter 4

Experiments and discussion

In order to evaluate the performance of the method presented in this paper we measure the normalized correlation between the predicted feature \( \hat{f} \) and the expected feature \( f \) as follows;

\[
\text{score} = \frac{\langle \hat{f}, f \rangle}{\| \hat{f} \| \cdot \| f \|}.
\] (4.1)

The Sparsity of the neural network components is measured by adding the \( L_0 \) norm of all the \( \text{Eigenbrains} \) within the \( E_{\text{brain}} \) subspace.

\[
\text{Sparsity} = \sum_{j=1}^{m} \| e_j \|_0
\] (4.2)

The Relative Sparsity is computed to get a sense of how sparse the low-dimensions subspace is in comparison with the total number of voxels \( p \) for the whole 3D image \( (p = 64 \times 64 \times 34) \), so, we define:

\[
\text{Relative Sparsity} = \frac{\text{Sparsity}}{m \times p}.
\] (4.3)

The 10-fold cross validation is used to weight and evaluate all the prediction scores. It separates the data into ten subsets with \( n/10 \) observations in each of them. A single subset is retained for testing of the model, and the remaining nine subsets are used as training data.
4.1 Design parameters: score vs. sparsity

Several parameters should be weighted in order to get accurate sparse Eigenbrains components at the of the Sparse Augmented Algorithm (ASPCA). The Decoding Neuronal Network approach also needs to be tuned by assigning values to the thresholds $\lambda_1$, $\lambda_2$ and $\lambda_3$.

For the studied values within the range for $\text{threshold}_1$, $\text{threshold}_2$ and $\text{threshold}_3$ we achieve the sparsity shown in Fig 4.1 in the $E_{brain}$ subspace composed with 14 Eigenbrains. In the graph (b) the y-axes is expanded for ($\text{threshold}_3 = 2, \ldots 11$).

In order to fine tune the semi-supervised neuronal decoding approach, we track the feature predictions using 10-fold cross validation of the features with better performance (Body parts - 5, Faces - 7, Language - 9 and Motion - 11) in the prediction model from section 3.3. The optimal design parameters consider a trade-off between the sparsity of $e_j$ and the prediction score.

In Table 4.1 we summarize the values with the maximum prediction score, nonetheless, in Table 4.3 we relax the final score by achieving more sparse and uncorrelated networks.
Figure 4.2: Prediction score as a function of threshold 1($\lambda_1$), thresholds 2($\lambda_2$), and threshold 3($\lambda_3$).

Table 4.1: Design parameters for maximum prediction score

<table>
<thead>
<tr>
<th>Feature</th>
<th>Prediction score</th>
<th>Sparsity</th>
<th>$th_1-th_2$</th>
<th>$th_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body Parts</td>
<td>0.8339</td>
<td>3321</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>Faces</td>
<td>0.7483</td>
<td>1181</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Language</td>
<td>0.9138</td>
<td>565</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Motion</td>
<td>0.9201</td>
<td>2328</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

4.2 Eigenbrains

Some of the sparse neuronal networks from the optimal values for the feature *Language* are collected in Fig 4.3. These are considered the more interpretable networks, having more aggressive...
Table 4.2: Sparsity details for maximum prediction score

<table>
<thead>
<tr>
<th>Feature</th>
<th>$|e_1|_0$</th>
<th>$|e_2|_0$</th>
<th>$|e_3|_0$</th>
<th>$|e_4|_0$</th>
<th>$|e_5|_0$</th>
<th>Rel.Sparsity</th>
<th>$th_1-th_2$</th>
<th>$th_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body Parts</td>
<td>1184</td>
<td>620</td>
<td>477</td>
<td>418</td>
<td>316</td>
<td>0.72%</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>Faces</td>
<td>485</td>
<td>257</td>
<td>213</td>
<td>138</td>
<td>43</td>
<td>0.26%</td>
<td>3</td>
<td>2</td>
</tr>
<tr>
<td>Language</td>
<td>338</td>
<td>125</td>
<td>76</td>
<td>26</td>
<td>0</td>
<td>0.12%</td>
<td>5</td>
<td>1</td>
</tr>
<tr>
<td>Motion</td>
<td>388</td>
<td>388</td>
<td>388</td>
<td>388</td>
<td>388</td>
<td>0.51%</td>
<td>1</td>
<td>1</td>
</tr>
</tbody>
</table>

Table 4.3: Design parameters for optimal neuronal networks architecture

<table>
<thead>
<tr>
<th>Feature</th>
<th>Prediction score</th>
<th>Sparsity</th>
<th>$th_1-th_2$</th>
<th>$th_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body Parts</td>
<td>0.8068</td>
<td>2525</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>Faces</td>
<td>0.7184</td>
<td>1166</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>Language</td>
<td>0.8639</td>
<td>976</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Motion</td>
<td>0.8973</td>
<td>1157</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>

Table 4.4: Sparsity details for optimal neuronal networks architecture

<table>
<thead>
<tr>
<th>Feature</th>
<th>$|e_1|_0$</th>
<th>$|e_2|_0$</th>
<th>$|e_3|_0$</th>
<th>$|e_4|_0$</th>
<th>$|e_5|_0$</th>
<th>Rel.Sparsity</th>
<th>$th_1-th_2$</th>
<th>$th_3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Body Parts</td>
<td>1037</td>
<td>480</td>
<td>353</td>
<td>273</td>
<td>188</td>
<td>0.55%</td>
<td>5</td>
<td>9</td>
</tr>
<tr>
<td>Faces</td>
<td>625</td>
<td>234</td>
<td>138</td>
<td>91</td>
<td>41</td>
<td>0.25%</td>
<td>6</td>
<td>4</td>
</tr>
<tr>
<td>Language</td>
<td>601</td>
<td>170</td>
<td>68</td>
<td>72</td>
<td>43</td>
<td>0.21%</td>
<td>9</td>
<td>6</td>
</tr>
<tr>
<td>Motion</td>
<td>669</td>
<td>205</td>
<td>107</td>
<td>86</td>
<td>55</td>
<td>0.25%</td>
<td>8</td>
<td>6</td>
</tr>
</tbody>
</table>

thresholds, as the prediction score is relaxed.

According to the Brodmann map, the auditory cortex (areas 41 and 42) which processes sound and contributes to our ability to hear, is selected in the Eigenbrain 2. The Eigenbrain 5 activates the areas 17,18,19 in the back of the brain that are associated with the visual cortex (V1,V2 and V3). The frontal eye fields(area 8) and the prefrontal cortex (area 9), involved with eye movements, are shown in Eigenbrain 4. See Fig 4.3.

In Fig 4.4, $e_2 = \text{yellow}$,$e_3 = \text{pink}$, $e_4 = \text{green}$ and $e_5 = \text{red}$, from the feature Language and $\text{threshold}_1$ and $\text{threshold}_2 = 9$ and $\text{threshold}_3 = 6$, are plotted over the same functional magnetic images. It can be inferred that the principal components are largely uncorrelated because they barely overlap. Indeed, they seem to define different functional modules.

In Fig 4.5 and Fig 4.6, the first six optimal architectural neuronal networks, for the feature Faces are pictured. The label in this case are: $\text{threshold}_1$ and $\text{threshold}_2 = 3$ and $\text{threshold}_3 = 2$. 
Figure 4.3: Optimal EigenBrains network architecture for Language threshold\(_1\) and threshold\(_2 = 9\) and threshold\(_3 = 6\). (a) EigenBrain 2 - slices 13/14/15 (b) EigenBrain 4 - slices 8/9/10 (c) EigenBrain 5 - slices 7/8/9. Slices are numbered from the top to the bottom

4.3 Prediction performance

Fig 4.7 shows the prediction scores for several values of sparsity for the features Body Parts, Faces, Language, and Body Parts; for \(m = 14\).

We show in Fig 4.5 that the first Eigenbrain selects almost all the voxels in the brain so it can be considered total brain activation. To study how Eigenbrain 1 (\(e_1\)) affects to prediction (see Fig. 4.8 and see Fig. 4.9) we project \(X\) into the regular subspace \(E_{\text{brain}}\) and onto \(E_{\text{brain}}^1\). In some cases it performs better without projecting onto \(e_1\).

\[ E_{\text{brain}}^1 = \begin{bmatrix} e_2(1) & \cdots & e_m(1) \\ \vdots & \vdots & \vdots \\ e_2(p) & \cdots & e_m(p) \end{bmatrix} \]
Figure 4.4: Overlap of $e_2$ (yellow), $e_3$ (pink), $e_4$ (green) and $e_5$ (red) for subspace $m = 14$
Figure 4.5: Optimal EigenBrains network architecture for Faces $\text{threshold}_1$ and $\text{threshold}_2 = 3$ and $\text{threshold}_3 = 2$. (a) EigenBrain 1 (b) EigenBrain 2 (c) EigenBrain 3 (d) EigenBrain 4
Figure 4.6: Optimal EigenBrains network architecture for Faces threshold$_1$ and threshold$_2 = 3$ and threshold$_3 = 2$. (e) EigenBrain 5 (f) EigenBrain 6

Figure 4.7: Kernel Ridge Regression Prediction score for $m = 14$
Figure 4.8: Prediction scores vs Sparsity for subspaces $E_{brain}$ and $E_{brain}^1$. (a) Body Parts (b) Faces
Figure 4.9: Prediction scores vs Sparsity for subspaces $E_{brain}$ and $E^1_{brain}$. (c) Language (d) Motion
Chapter 5

Discussion and conclusion

In this thesis, we propose a new method for the analysis of fMRI data. Our approach discovers clustered brain areas by using sparse PCA representations without any spatial constraint in the optimization problem. We do not apply any spatial smoothing function in addition to the sparse coordinate selection, however, the principal components appear to be concentrated around specific areas in the cerebral cortex, which are really helpful in the interpretation stage.

We have proposed a technique to identify a low dimensional neuronal network subspace $E_{brain}$, where all the principal components are $p$-dimensional vectors ($\in \mathbb{R}^p$), also called $Eigenbrains$. Only some few entries from each of them are different than 0, and the sparsity is controlled by $\lambda_1$, $\lambda_2$ and $\lambda_3$. We allow the $E_{brain}$ to be different for each task developed in the brain. Within an specific subspace, the sparsity increases according to the index of the $Eigenbrain$ ($\|e_1\|_0 > \|e_2\|_0$).

The adaptive combination of $Eigenbrains$ defines the brain activation for a specific feature. The combination of $Eigenbrains$ can be interpreted according to the massive modularity hypothesis. Each $Eigenbrain$ can be understood as a small number of well defined functional areas (some of them coincide with Brodmann areas).

It is worth noticing the efficiency of this semi-supervised method. With a small number ($m = 14$) of extremely sparse $Eigenbrains$ and relative sparsity values in the range from 0.72% to 0.12% we achieve final prediction scores ranging from 0.7 to 0.92 for the studied features. This implies a really small computation time cost. The approach performs competitively in terms of prediction score compared with other submissions to the PBAIC contest. As a general rule, the
more components we include in the new low-dimensional subspace, the more variance is captured and the better the performance is in terms of prediction, nevertheless, there exists the special behavior of the first *Eigenbrain*.

The model obtained with this approach is interpretable because the input fMRI data is very rich in external stimulus. On the other hand, the features corresponding with expected diffuse brain activation, such as self-awareness, might not be well predicted.
Bibliography


[28] [Online]. Available: http://www.fil.ion.ucl.ac.uk/spm/software/spm8/

