Automatic Classification of Cognitive States November 2010

Carlos Alberto Falé Cabral

Universidade Técnica de Lisboa
Instituto Superior Técnico, Lisboa Portugal
carlos.fale@gmail.com

Functional Magnetic Resonance Imaging has established itself as the most powerful technique available today to measure brain activity induced by a perceptual or cognitive state. The inverse problem is considered in this study; given the measured brain activity, our goal is to predict the perceptual state. Machine Learning algorithms were used to address this problem in this work. Multi-subject fMRI data analysis poses a great challenge for the machine learning paradigm, by its characteristics: the low Signal to Noise Ratio (SNR), high dimensionality, small number of examples and inter-subject variability. To address this problem, several methods of classification and feature selection were tested. The main criterion of feature selection was mutual information in a univariate method, but a multivariate feature selection was also proposed. Both a single classifier and an ensemble of classifiers were tested. The ensemble of classifiers approach consisted on training an optimized classifier for each class and then the combination was made. The data analysed was obtained from three multi-subject experiments of visual stimulation with 4 classes of stimuli, at different magnetic field strengths. The ensemble of classifiers performs best for most data sets and methods of feature selection. The multivariate method does not show overall improvement in the classification. In summary, the results suggest that a combination of classifiers can perform better than a single classifier, particularly when decoding stimuli associated with specific brain areas.

***Index Terms*—Brain decoding, ensemble of classifiers, fMRI, machine learning, multivariate feature selection, retinotopic mapping, and visual localizer.**

I. Introduction

Brain Imaging is nowadays one of the most exciting fields in neurosciences, as it offers the possibility of measuring brain activity in awake human subjects *in loco* [1]. These measurements lead to the development of brain mapping methods, associating perceptual or cognitive states with spatial or temporal patterns of brain activity.

Neuroimaging techniques for measuring brain activity include methods as different as electro-encephalogram (EEG), positron emission tomography (PET), magneto-encephalogram and functional Magnetic Resonance Imaging (fMRI). The later is probably the most common method of assessing brain activity in humans due to the good compromise between its temporal and spatial resolutions as well as its completely non-invasive nature.

In each fMRI experiment, Blood Oxygen Level Dependent (BOLD) signals are recorded while the subject performs a task or experiences a stimulus [1]. The identification of the active brain regions in response to the experimental manipulation relies on detecting the differences in BOLD signal significantly correlated with the experimental paradigm. These differences are small, on the order of 5%, hence the signal to noise ratio (SNR) is intrinsically low for this technique.

The usual approach to detect the active brain regions is statistical analysis; this statistical analysis is carried out using a linear approach by a General Linear Model (GLM) that takes into account the experimental manipulation and any existing confound variables. Therefore for each voxel the model is adjusted and a 3-D map of parameters estimates is created [2]. The active patterns are then determined by using the appropriate inference procedures.

In the last few years, driven by the increasing number of available data and the advent of new machine learning techniques, there has been a growing interest in the application of machine learning algorithms to fMRI analysis [3, 4]. This interest is supported by studies that demonstrate the possibility of extraction of new information from neuroimaging data [5, 6].

While the established methods of fMRI analysis, like GLM based analysis, look to find the brain activity pattern that corresponds to a stimulus or task, in machine learning classification analyses the question is inverted and the goal is to find the stimulus or task that correspond to the recorded brain activity pattern. Although this is the most common scientific question in fMRI machine learning classifier analysis and by extension the main focus of this study, there are other relevant questions that can be posed, especially in what regards activation patterns. For example whether there is information about a variable of interest, (pattern discrimination; where is the information is, (pattern localization) and how the information is encoded, (pattern characterization) [7]. Although these questions are not the main focus of this study, they will not be forgotten and will be addressed, as they can provide important information about the brain inner organization and function.

The visual cortex shows functional differentiation as distinct stimulus induce specific brain activation patterns [8]. Primary visual cortex is one of the best examples of such organization; it is well documented that this structure shows retinotopic organization. Each visual quadrant field, and therefore retina, is mapped in a well defined region in the primary visual cortex. The stimulus suffers double inversion, first it is flipped upside down in the retina and the sides inverted in the optic chiasm. Another example of the human brain specialization is the different response patterns observed for distinct categories of visual stimuli like faces, houses or tools [9]. These patterns consist of a network of brain regions that are differentially activated according to the presented stimuli. In order to identify these specific patterns, fMRI experiments were conducted consisting in the alternated presentation of different categories of visual stimuli, such as faces, houses, objects or scrambled objects. This kind of experience is therefore denominated localizer experience. It was found that a small area in the fusiform cortex responds to the stimuli faces more than to any other, the so called face fusiform area (FFA). It was also observed that a region in parahhipocampal cortex activates more for houses than for faces or other objects, the parahhipocampal place area (PPA). When recognized objects images are compared with unrecognized, scrambled images of the same objects, a large are in the lateral occipital cortex (LOC) shows greater activation.

A. BOLD signal

The signal measured in an fMRI experiment, BOLD, has unique spatio-temporal properties. The BOLD signal has a temporal delay of 3-4 seconds associated with the neuronal underlying process. The theoretical representation of the BOLD signal response to an impulse stimulus is known as Haemodynamic Response Function, HRF. The BOLD signal also yields an intrinsic spatial blurring, in other words, neighbour voxels have highly correlated signals across time.

B. Machine Learning Overview

Machine Learning is a scientific discipline that is concerned with the production and study of algorithms with the ability to learn and predict behaviours and patterns from data. The definition for machine learning by Tom Mitchell [10] is probably the most accepted and states that "A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P, if its performance at tasks in T, as measured by P, improves with experience E". In the particular problem presented in this work, the interest falls over the sub-area of supervised learning. Supervised learning deals mainly with regression and classification problems; classification is the discrete analogue of regression in which the predicted variable assumes discrete values.

The classification problem, in a more formal way, can be defined as given a distribution of labelled examples $\mathcal{X}$ randomly withdraw an example $\mathbf{x}$ and classify it correctly.

In order to obtain a classification, a function that can learn and map characteristics from the distribution to the example set to classify is required. Such a function that receive as input an example and returns a label or category is called a classifier,

$$\hat{y} = f(\mathbf{x}) \tag{1}$$

where $\hat{y}$ is the predicted class label, f the classifier and $\mathbf{x}$ the example to classify.

The problem of assessing the performance of a classifier can be defined as the expected value of the classification error for an example $\mathbf{x}$ randomly withdrawn from the example distribution $\mathcal{X}$. The problem definition gives rises to immediate issues, first, in practice, the data set has a finite number of examples and classifiers cannot test and train with the same data. To approach this limit situation the cross validation procedure is used. In cross validation the distribution is divided in k equal parts, then the classifier trains with k-1 parts, training set, and test with the remainder one, testing set. This procedure is repeated for all the k parts, (k-fold cross-validation) and the mean classification error is assumed as the error of the classifier. It is of the utmost importance that there are not any kind of information flow between training and test sets in order to obtain a good estimation of the expect classification accuracy.

A good way to assess how well a classifier is performing is to compare its accuracy with the expected accuracy for the chance classifier. A chance classifier is a classifier that learnt nothing and so its estimate is equivalent to a random guess, for instance, in a two class classification the expected accuracy for the chance classifier is 50%, for a four class it is 25% and so on.

C. State of the art

The fMRI data analysis with machine learning algorithms poses a great challenge due to low SNR, small number of examples and high dimensionality of the problem. Hence feature selection processes became mandatory. Several feature selection methods are available for this particular problem. The *filtering/scoring* methods are among the most used; these methods rank the features according to a given criterion and the best ranked features are chosen. The most common methods consider the *n* most active, most discriminatory, with the highest search light accuracy or most stable voxels in the whole brain or in regions of interest, ROI, as well as the average of these criterions across a ROI [3, 7, 11, 12] . *Wrapper* methods have already been used [13]. These methods use a learning machine to evaluate sub sets of features and decide the feature selection by the impact of the new features in the previous selected features [14], however due to high computational costs their use is sometimes prohibitive.

The classifiers more used in the fMRI analysis with machine learning algorithms are Support Vector Machines (SVM), k Nearest Neighbours (kNN), Gaussian Naive Bayes (GNB) and Fischer Linear Discriminant (FLD) [3, 7, 12, 15] . Although SVM is the most powerful classifier from this set of classifiers it is the one with greater computational costs. The choice of the best classifier for a given experiment is not clear and so several classifiers are tested and the best performing chosen.

However it was observed in several cases, especially high dimensional, that an ensemble of classifiers performs better than the single classifier approach [16, 17]; the use of ensembles of classifiers is a poorly explored area in machine learning fMRI analysis. Nevertheless some studies have been made using ensemble of classifiers in fMRI data. In [18] ensembles of decision trees were used to decode connectivity in fMRI and, in [19], several ensembles of different classifiers were tested but only for single subject data. In multi-subject studies the validation procedure is usually done by leave-one-subject-out cross validation. In other words each subject data corresponds to one fold [3, 7, 12]. The feature extraction process can be made in very different ways. It is common to average images within a block but the utilization of a single image was also done [3, 7, 12]. Ultimately the choice of the feature extraction method depends on the data.

D. *Objectives*

The objective of the work presented in this Thesis is to develop machine learning methodologies to decode the stimulus presented to a subject at a given time point during a visual mapping experiment. In particular, the feature selection approaches, the type classifier and the possibility of combining sets of classifiers will be investigated.

II. Materials and Methods

A. *Materials*

1) *Data sets*

The first data set correspond to the mapping of each of the four different quadrants of the primary visual cortex to assess the retinotopic organization of this structure (*Mapping Experiment)*. The stimulus consisted on four black and white checkerboards wedges flashing at 8 Hz as represented in Fig.1.

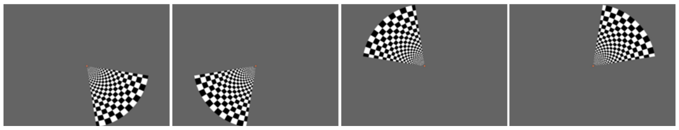
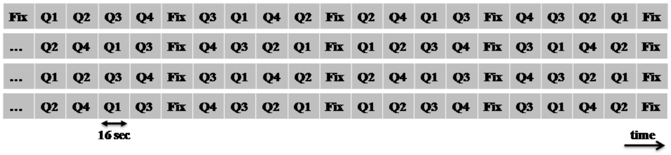


Figure 1 Stimuli used in the *Mapping Experiement*, checkerboard wedges corresponding, from left to right, Q1, Q2, Q3 and Q4 [20]

The paradigm consisted on a block design with the four stimuli alternating with the fixation in 16 seconds block in the order shown in Fig. 2.



Fix	Q1	Q2	Q3	Q4	Fix	Q3	Q1	Q4	Q2	Fix	Q2	Q4	Q1	Q3	Fix	Q4	Q3	Q2	Q1	Fix
...	Q2	Q4	Q1	Q3	Fix	Q4	Q3	Q2	Q1	Fix	Q1	Q2	Q3	Q4	Fix	Q3	Q1	Q4	Q2	Fix
...	Q1	Q2	Q3	Q4	Fix	Q3	Q1	Q4	Q2	Fix	Q2	Q4	Q1	Q3	Fix	Q4	Q3	Q2	Q1	Fix
...	Q2	Q4	Q1	Q3	Fix	Q4	Q3	Q2	Q1	Fix	Q1	Q2	Q3	Q4	Fix	Q3	Q1	Q4	Q2	Fix

Figure 2 Paradigm used in the Mapping Experiment with block of four stimuli alternating with fixation [20]

The data was obtained from 5 healthy subjects in two different sessions each on a 1.5T Philips system using BOLD image to collect 672 volumes with TR=2000ms, 24 slices and a voxel resolution of 3.750x3.750x5.000 mm^3, yielding an image size of 64x64x24.The second and third data sets come from visual localizer experiments (*Localizer Experiment1, Localizer Experiment2)*. These visual localizer experiments look to find the areas in the visual cortex specialized in four visual stimuli: Faces, Houses, Objects and Scramble, represented in Fig. 3.

The paradigm for the localizer experiments is very similar to the previous one consisting of different ordered, sequences of the four stimuli interspersed by fixation periods on a block design with each block having 18 seconds, as explicit in Fig. 4.

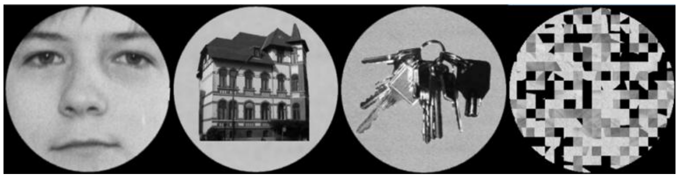
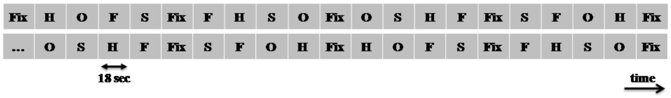


Figure 3 Stimuli used in Localizer Experiment1 and Localizer Experiment 2, from the left to the right Faces, Houses, Objects and Scramble [21]

Localizer Experiment 1 was obtained from 10 healthy subjects in two sessions, on a 3.0 Philips system using BOLD imaging to collect 118 brain volumes with TR=3000ms, 38 slices and a voxel resolution of 2.875x2.875x3.000 mm^3 yielding an image size of 80x80x38. *Localizer Experiment 2* was obtained from 10 healthy subjects in one session, on a 7.0 Siemens system using BOLD imaging to collect 112 brain volumes with TR=3200ms, 40 slices and a voxel resolution of 2.019x2.019x2.000 mm^3 yielding an image size of 104x104x40 .



Fix	H	O	F	S	Fix	F	H	S	O	Fix	O	S	H	F	Fix	S	F	O	H	Fix
...	O	S	H	F	Fix	S	F	O	H	Fix	H	O	F	S	Fix	F	H	S	O	Fix

Figure 4 Paradigm used in the Localizer Experiment1 and Localizer Experiment2 with blocks of four stimuli alternating with fixation [21]

Localizer Experiment 1 and *Localizer Experiment 2* correspond to the same experience; however the different acquisition properties result in differences in the data consistency.

In all the experiences, for co registration and anatomical reference purposes it was acquired for each subject a high resolution structural image using a T_1 weighted imaging sequence.

2) *Pre-Processing*

Several pre-processing options are necessary before the data is fit to be fed to a classifier; before the feature extraction it is necessary to make the standard fMRI pre-processing procedure.

The datasets were independently processed and analyzed using the FSL software package (http://www.fmrib.ox.ac.uk/fsl).

The following pre-processing steps were performed on each BOLD time series: motion correction [22]; non-brain removal [23]; mean-based intensity normalization of all volumes by the same factor; spatial smoothing (Gaussian kernel, 5 mm FWHM) and high-pass temporal filtering (Gaussian-weighted least squares straight line fitting, 50 sec cut-off).

As all the data sets are multi subject, registration became necessary; to perform registration the FLIRT tool from the FSL [22] . The reference image used for registration was the MNI standard available in FSL package with voxel definition of 2.000x2.000x2.000 mm^2 [24]. For the *Mapping Experiment* and *Localizer Experiment 1* data sets i*n* order to avoid the unnecessary increase of dimension the MNI standard was re-sampled to the data sets resolution, for the *Localizer Experiment 2* data set it was used the MNI standard was used since they have similar resolution.

The functional images were first registered to the correspondent high resolution structural image using an affine rigid body transformation with 6 degrees of freedom. The structural image was registered to the standard MNI image with an affine transformation with 12 degrees of freedom, and the composition of the two transforming matrices was applied to the functional image to register them in MNI space.

To access data consistency, after pre-processing, all the runs from all the datasets were analyzed using the FSL software to find patterns of activation by a common use GLM analysis. In this process two subjects were found to have incoherent activation patterns, one for the *Mapping Experiment* and another for the *Localizer Experiment 2*. Therefore these two subjects were excluded. In the end the *Localizer Experiment 1* had the same number of subjects, 10; while the *Mapping Experiment* and the *Localizer Experiment 2* number of subjects was reduced to 4 and 9 respectively.

To identify the voxels that correspond to brain matter, the MNI atlas was used. Besides this approach all the voxels that have zero value across all the examples were ignored. In the end the *Mapping Experiment* yields a total of 512 examples each on with 27851 features; the *Mapping Experiment 1* has 320 examples each on with 60922 features and the *Mapping Experiment 2* yields 144 examples with 123494 features each one.

B. *Methods*

1) *Classifiers used*

Three different classifiers were used in this study, GNB (Gaussian Naive Bayes), kNN (k-Nearest Neighbours) and SVM (Support Vector Machines). In addiction ensemble of classifiers for kNN and GNB were also used. All the classification methods, (kNN, GNB and ensembles) were implemented in MatLab® except for the SVM for which the toolbox LibSVM [25] was used.

Let x=$x_1, .. x_n$ be a pattern, where n represents the number of features, in our problem and ω_j, j=1,...c, denotes the classes associated to different visual stimuli.

a) *GNB classifier*

The Gaussian Naive Bayes classifier is a probabilistic classifier based on the Bayes rule that has strong independence assumptions about the Gaussian distribution of the features. In other words, the classifier assumes that the features are independent, which accounts for the term naive in the name, and came from a Gaussian or a Gaussian mixture distribution. The final output of the GNB classifier is the conditional probability of the example **x** belonging to the class ω_j, for j=1,...c, knowing **x**. According to Bayes rule this probability can be written as:

$$P(\omega_j|\mathbf{x}) = \frac{P(\omega_j).\prod_i P(x_i|\omega_j)}{\sum_k P(\omega_k).\prod_i P(x_i|\omega_k)} \qquad (2)$$

b) *kNN classifier*

The kNN classifier is a very simple non-parametric classifier. This classifier treats examples as vectors in a feature space and the classification is determined by the most common class of the k nearest examples in the training set. So when an example **x** is presented to the kNN classifier, it will determine the nearest k examples in the training set and classify as the most common class among this sub-set. The distance measure can be chosen from a wide range of metrics like Euclidean, Mahalanobis or Minkowski.

The output of this classifier, unlike the GNB, is a class and not a set of normalized probabilities for each class. This is indeed indispensable for the usage of an ensemble of kNN classifiers and its combination. To overcome this problem the distance based probability as proposed in [17] for the kNN classifier was implemented.

$$\hat{P}(\omega_j|\mathbf{x}) = \frac{\sum_{\mathbf{x}^j \epsilon \omega_j} \frac{1}{d(\mathbf{x},\mathbf{x}^j)}}{\sum_{\boldsymbol{i=1}}^{\boldsymbol{k}} \frac{1}{d(\mathbf{x},\mathbf{x}^i)}} \qquad (3)$$

c) *Classifier Ensemble*

In this study, in addiction o the single classifiers presented in the preceding section, two ensembles of classifiers were also used, one for kNN and one for GNB. The underlying idea is to optimize a classifier for each class, corresponding to a stimulus, and then combine the results of each classifier.

The degree of support from a classifier for an example input **x** can be defined in different ways; in this study the estimates of the posterior probabilities for the classes were used.Both GNB and kNN formulations regard the possibility of posterior probability estimation and so are fit for the classifier ensemble methods proposed.

The combination of classifiers can be done in several ways; the first step to the combination is the construction of a decision profile (DP(**x**)). Using the same notation as for the previous classifiers, let $\mathbf{x} \in \boldsymbol{\Re^n}$ be a feature vector and $\Omega = \{\omega_1, \omega_2, \dots, \omega_c\}$ be the set of class labels. Each of the classifiers D_i in the ensemble $\mathfrak{D} = \{D_1, D_2, \dots, D_L\}$ returns c degrees of support, that in this case are probabilities and so $D_i: \boldsymbol{\Re^n} \rightarrow [0\ 1]^c$. If the support that a classifier D_i gives to hypothesis that **x** come from the class ω_j is defined as a point position $d_{i,j}$ in a matrix it is possible to build a decision profile like in (4). The lines of the matrix have sum one, and the columns represent the support given by each classifier to that class.

$$DP(\mathbf{x}) = \begin{bmatrix} d_{1,1}(\mathbf{x}) & \cdots & d_{1,c}(\mathbf{x}) \\ \vdots & \ddots & \vdots \\ d_{L,1}(\mathbf{x}) & \cdots & d_{L,c}(\mathbf{x}) \end{bmatrix} \qquad (4)$$

In this work each classifier in the ensemble will be specialized for each class therefore the number of classifiers L=c, and $DP(\mathbf{x})$ a square matrix.

Five different methods for classifier combination where tested in this study, four non trainable and one trainable. Trainable means that the classifier combiner needs to learn parameters from the data. For the non trainable methods of classifier combination the support given by the ensemble of classifiers to class ω_j is:

$$\mu_j = \mathcal{F}[d_{1,j}., .., d_{L,j}] \qquad (5)$$

where $\mathcal{F}$ is a combination function and the class label of **x** is the index of the maximum of μ in equation (5).

Comb 1 - Assuming that each classifier is trained in order to be optimized for a given class the relevant information is localized in the diagonal of the decision profile, this method of classifiers combination simply considers that the support given by the ensemble of classifiers to a class ω_j is $d_{j,j}$.

Comb 2 - Simple mean, $\mathcal{F}$ =average.

Comb 3 - Maximum, $\mathcal{F}$ =maximum

Comb 4 - Median, $\mathcal{F}$ =median

Comb 5 - This combination method is the only trainable method used in this study. The underlying idea is to find a weight for each of the positions of the decision profile with the constraint that the sum of the weights along a column be equal to one. This can be formulated as a regression problem where the target values are the posterior probabilities $P(\omega_j|\mathbf{x})$**.** In the classification problems the posterior probabilities are 1 if the example has class label ω_j and 0 otherwise. Considering a noise free problem the expected posterior probability given by a classifier would be the same binary combination 0 or 1. So for class label ω_j, the support given by each classifier is subtracted to the expected posterior probabilities. Hence for each class ω_j there is an error matrix with L lines and m columns. The goal now is the error variance minimization, however is important to stress that this error is not the classification error but the approximation error. The next step is the computation of the covariance matrix for the approximation error matrix. If it is assumed that the expected error in approximating the posterior probability $P(\omega_j|\mathbf{x}) - d_{i,j}(\mathbf{x})$ is normally distributed with mean 0, the problem can be solved by *constrained regression.* By minimizing the function (6), which already includes the Lagrange multipliers to constrain the sum of the weights of each column to be one. Where σ_{ij} denotes the covariance between the classifiers D_i and D_k.

$$J = \sum_{i=1}^{L}\sum_{k=1}^{L} w_i w_j \sigma_{ik} - \lambda\left(\sum_{i=1}^{L} w_i - 1\right) \qquad (6)$$

A solution for this minimization problem is:

$$w = \Sigma^{-1}\boldsymbol{I}(\boldsymbol{I}^T\Sigma^{-1}\boldsymbol{I})^{-1} \qquad (7)$$

where w is weights vector, Σ is the covariance matrix and L is a L-element vector with ones [17]. The procedure of coefficients estimation was made in a nested cross validation procedure using leave-one-subject-out cross validation. For the nested cross validation the classification errors were averaged for all folds and the coefficients estimated.

2) *Feature Extraction*

The example creation, usually includes a feature extraction step where block image combination in order to reduce the noise. In this work four different methods to perform the block image combination were used.

Mean – The most usual method to combine images in a block consists in using simply the mean image.

Central Image – In this combination method only a single image from the block is used, the central image, if there is a pair number of images in a block the mean of the two central images is used.

Gaussian Window – This method corresponds to the application of a Gaussian window to the image block, in practice it is a weighted mean in which the coefficients are driven from a Gaussian window using the MatLab ® implemented formulation [26]:

$$g[h] = e^{-\frac{1}{2}\left(\beta\frac{h}{H/2}\right)^2} \qquad (8)$$

where $-\frac{H}{2} \le h \le \frac{H}{2}$ and $\beta \ge 2$, and the length of the window is T=H+1. The β parameter is the reciprocal to the standard deviation.

Custom Window – The BOLD signal has well known temporal characteristics. In order to apply this information in the feature extraction process a custom window was idealized for this particular problem. Each image is acquired with an interval, TR, in this study 2 and 3 or 3.2 seconds, the delay of the haemodynamic response is about 3-4 seconds. Considering that the stimulus presented do not cause physiological habituation it was defined the mean of the block without the first image as feature extraction method.

After the block image combination process, for each run the signal was transformed into Percentage Signal Changes, *PSC,* relative to the mean value for the baseline condition and as final procedure the PSC was scaled in order that of features have mean 0 and variance 1. All the process of feature extraction was performed for each fMRI run independently in order to reduce inter run differences and make sure that there is total separation between training and testing set.

3) *Feature Selection*

The feature selection process is a crucial step for the classification success due mainly to the noisy data and the high dimensionality of the problem. In this section the used methods to perform feature selection are described.

a) Univariate Feature Selection Method and Search Light Algorithm

The main criterion for feature selection used in this study was Mutual Information. Mutual Information measures how much information two variables share; hence a feature is more important if the mutual information between their labels and that feature distribution is larger [14]. Let $\mathbf{f}_i$ be an *m-dimensional* vector that contains all the examples for feature x_i and y a label vector with the same size as x_i the class labels vector then,

$$MI(\mathbf{f}_i, y) = H(\mathbf{f}_i) + H(y) - H(\mathbf{f}_i, y) \qquad (9)$$

where *H(.)* is the entropy of a random variable and *H(.,.)* represents the joint entropy. Mutual Information range of values is between -1 and 1.

Two other criteria for feature ranking were also implemented, accuracy and the search light accuracy method**,** in order to assess the performance of mutual information as a feature selection method for this particular problem.

The considered feature selection methods rank the features according to a particular criterion but do not give any information about the number of features or threshold for the criterion to use, as they are *scoring/filtering* methods. The *a priori* definition of a number of features or threshold in MI to be used based on the experience is a usual procedure. Although simple and computationally efficient it assumes knowledge about the problem that cannot be extended to more general cases. Therefore it was desirable to estimate a number of features to use without a prior knowledge of the problem being addressed.

A common solution when there are parameters that need to be estimated is the use of nested cross validation inside the training set. This solution was applied to this particular problem. For the number of features estimation, *Number Estimation*, a range of values for the number of relevant features between 0.0005% and 10% of the total number of features was considered. This set was sampled in ten equally spaced points; for each point a nested cross validation procedure in the training set was applied and the accuracy assessed. The best performing number of features was identified and the procedure was repeated within the interval starting in the number of features considered before the maximum and ending in the next to the maximum considered value of features. The stop criterion was the variance of the interval classification being less than 1% or the number of features in interval less than 20. Other approach was the estimation of the MI threshold instead of the number of features, *Threshold Estimation*. In this work, the MI threshold is always relative to a fraction of the maximum MI for a given training set and vector of labels. The method was the same as the described for the estimation of the number of features, the initial interval considered was between the 0.0001 and 1 and a minimum threshold of the number of features was established at 10 to avoid overfitting.

The feature selection process for the ensemble of classifier is identical to the described previously, but as each classifier is optimized for a given class label, the mutual information is calculated for each $\mathbf{f}_i$ with a binary label vector that for classifier D_i is 1 when y_i is equal to ω_i and is 0 otherwise.

b) Multivariate Feature Selection Method

The proposed multivariate feature selection method was developed for the particular problem of feature selection in fMRI machine learning problems. The underlying idea is to take advantage of the spatial properties of brain structure and BOLD signal to find the most relevant features.

Wrapper methods for feature selection are most of the time prohibitive in what concerns computational cost and there is need to use *filter/scoring* methods. However hybrid *filtering/wrapper* methods have been developed for problems of high dimensionality [27]. The method proposed consists in combining the prior information about the problem spatial constrains and the *filtering* feature selection method to decrease the computational costs of the *wrapper* approach.

The method starts by selecting the best ranked feature and its 3-D neighborhood and classify it in a nested cross validation procedure. Then a face of the cube is grown one voxel and the new set is classified. If the classification accuracy improves the new set is taken and becomes the reference, if not, this growing direction is labeled as non relevant and the data set will not go further more in this direction. The procedure is repeated for all the directions, until all of them are considered of non relevant growing. The method is then repeated for the next best ranked features that were not selected yet. In the end there is a set of candidate regions labeled by its classification accuracy. The stopping criterion was that the numbers of different features in the regions to surpass 10% of the total number of features or a threshold in the total number of regions.

Regions were ranked its solo classification accuracy, taking the first one as the reference set of features and classifying by a nested cross validation, then joining the second region and classify, if an increase in classification accuracy is observed for the reunion of the two regions, this new set of features becomes the reference set of features, otherwise the first region is maintained as the reference set of features and the same for the rest of the candidate regions. In a more intuitive formulation, only regions that bring some classification benefit to the reference feature set are added.

III. Results

In this section, the results obtained with the previously described methods are presented. All the accuracy results were obtained using leave-one-subject-out cross validation procedure. For the GNB classifier $P(\omega_j|\mathbf{x})$ was modelled by a univariate Gaussian distribution and consequently μ_i and σi were estimated for each feature. The kNN classifiers used k=9 as number of neighbours and considering the Euclidean distance. The SVM classifier was only used as single classifier with a Gaussian kernel and parameters estimated by nested cross validation in the training set. For all the feature selection methods that required nested cross validation procedures the learning machine used was the same as the classifier in the final classification process, except for SVM due to the unaffordable computational cost.

A. Feature Scoring

The feature scoring method proposed, MI, was compared with two established methods: accuracy and the search light accuracy. Considering the 10% best scored features there is no significant differences between the three methods (results not shown) and since MI is much more efficient computationally this method is used in the rest of this work.

B. Feature Extraction

The four methods of image block combination for feature extraction were tested for all the three data sets with a fixed number of features. The best performing method was the *Custom window* closely followed by the *Gaussian Window* and *Mean*. The worst performing method is the *Central Image* (results not shown). Suggesting that the averaging process is important in the noise reduction; the first image is a transition image and introduces some noise to the example; and finally the last images as there is no physiological habituation to the stimulus contain useful information for the example construction process.

C. Classification methods

1) Single Classifier and Ensemble of Classifiers with fixed number of features.

In order to assess the performance of single classifier and ensembles of classifiers the classification accuracy for both classification methods was determined for several numbers of features, using both GNB and kNN. The combination in the ensemble was made using *Comb 1*. The results for all the three data sets and for kNN and GNB are shown in Fig. 5.

The results in Fig. 6 show that the maximum accuracy is achieved, for both *Localizer Experiment 1* (ensemble of kNN

82.81%) and *Localizer Experiment 2* (ensemble of kNN 77.08%), for the ensemble of classifiers. In *Mapping Experiment* the accuracy is near to 100% preventing comparisons due to a ceiling effect. In general, for all the three data sets, the ensemble of classifiers outperforms the corresponding single classifiers for a small number of features. Motivated by these results the single class accuracy for the ensemble of classifiers was determined; the results are represented in Fig. 6.

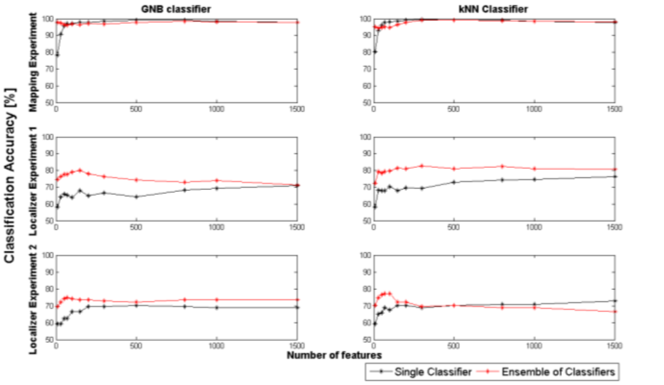


Figure 5 Classification Accuracy for various numbers of features for the three data sets for single and ensemble of kNN and GNB

The classification accuracy has different behaviours for different classes for the *Localizers Experiments*. This fact suggests that a method capable of selecting different number of features for the different classifiers and therefore different classes could achieve better results.

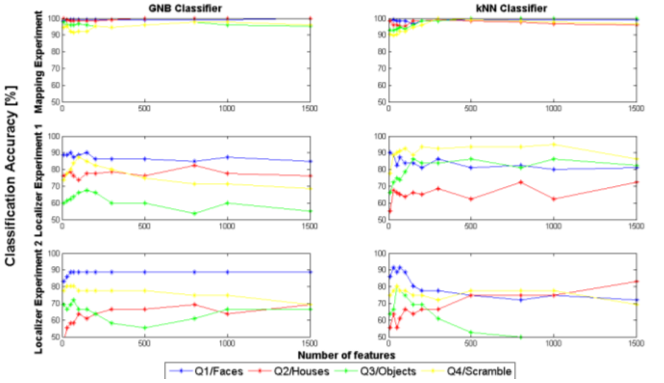


Figure 6 Classification accuracy for class for different number of selected features obtain by the ensembles of classifier for kNN and GNB, for all the data sets

2) Single Classifier and Ensemble of Classifiers with fixed MI threshold.

Using a threshold in MI is a solution for each classifier in the ensemble to choose a different number of features. The threshold is always a fraction of the maximum and so it is constrained to values between 0 and 1, this range of values was explored starting in 0.01 and 0.95 in intervals of 0.05.

In Fig. 7 the number of features selected as a function of the MI threshold for the *Localizer Experiment 1* is represented. As expected each class as a different profile of feature selection.

In Fig. 8 the classification results for different thresholds in MI are presented.

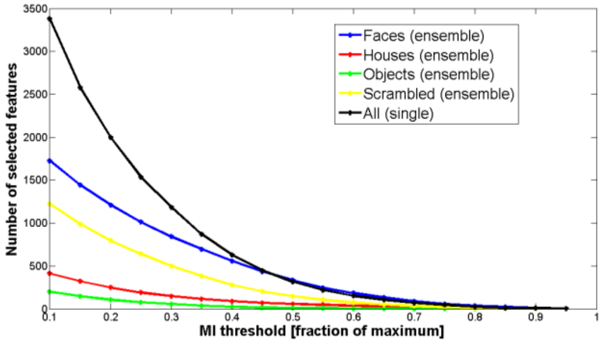


Figure 7 Number of features selected in function of the MI threshold for the *Localizer Experiment 1*

The same procedure used for the fixed number of features was then applied to the MI threshold and the results are shown in Fig 8.

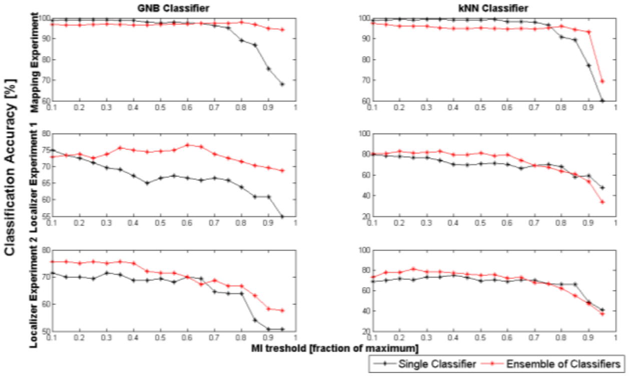


Figure 8 Classification Accuracy for different mutual information thresholds for the three data sets for single and ensemble of kNN and GNB

As observed in the previous section for both the *Localizer Experiment 1* (ensemble of kNN 82.81%) and *Localizer Experiment 2* (ensemble of kNN (81.25 %) the classification accuracy maximum is achieved for the ensemble of classifiers. More the ensemble of classifiers shows overall improvement over the single classifier for both classifiers, kNN and GNB. Again the results for the *Mapping Experiment* data set are of difficult interpretation due to the ceiling effect.

3) Single Classifier and Ensemble of classifiers with Estimated Number of features and MI Threshold

The successes of the previous results lead to the development of a method that could automatically choose a number of features or MI threshold. For the ensemble of classifiers five different combination methods were tested as described in II.B.1)c).

a) Mapping Experiment

The classification accuracy for the *Mapping Experiment* with both estimation methods for single GNB and kNN and ensembles of the same classifiers are shown in Table 1. The classifier combination represented is the one that performs best for this data set, *Comb 2*.

The ensemble of GNB outperforms the single GNB for both estimation methods. In what concerns kNN, with the *Threshold Estimation* method the ensemble of kNN performs as well as the single classifier and for *Number Estimation* method the ensemble of classifiers is outperformed. The fact that *Comb 2*, mean, is the best performing method suggests that more than a classifier has important information about a single class.

Table 1 Classification Accuracy for single classifier and ensemble of classifiers for *Mapping Experiment*

Accuracy (%)	*Number of features*		*Threshold in MI*	
	Best Fixed	*Estimation*	*Best Fixed*	*Estimation*
Single GNB	99.03	98.63	99.03	98.24
Ensemble of GNB	99.43	98.83	98.43	98.63
Single kNN	99.41	99.41	99.22	99.41
Ensemble of kNN	99.41	98.63	97.81	99.41

The features selected by the *Threshold Estimation* feature selection method for both single and ensemble of GNB are shown in Fig. 9. To avoid unnecessary repetition only a features map for the single classifier and ensemble of classifiers will be presented for each data set.

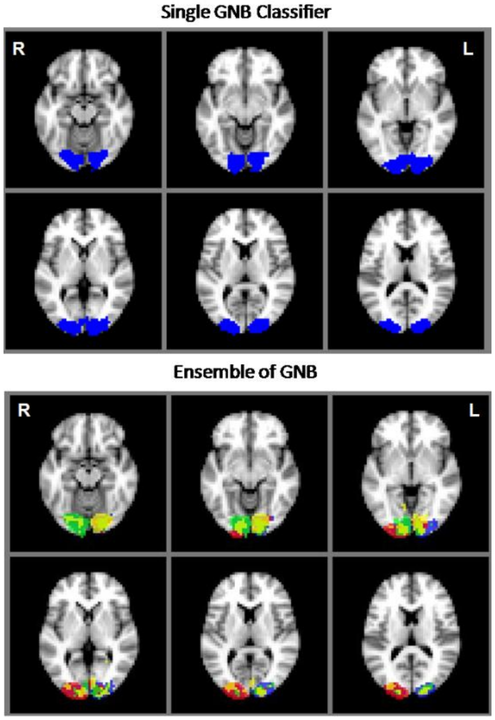


Figure 9 Voxels corresponding to the features selected for the *Mapping Experiment* by the *Threshold Estimation* method for GNB classifier

The features selected for each one of the estimation method and classifiers used as learning machines in the nested cross validation procedure are very similar. The features selected are in concordance with the expected as they are localized in the primary visual cortex. More, the features selected for the ensemble of classifiers, although the overlap between classes, reflect the retinotopic organization of the primary visual cortex. The overlap between the features selected for different classes is natural and expected for this experiment due to the vicinity of the activated zones. The performances of the estimation methods are satisfactory as the classification accuracy is very close to the maximum of the best fixed number approaches. The SVM classification for this data was as good as for kNN for the *Number Estimation* method (99.41%)c and worst than kNN for the *Threshold Estimation method* (99.02%).

b) Localizer Experiment 1

The results for the *Localizer Experiment 1* data set for the estimation methods and the best fixed number of features are represented in Table 2. The classifier combination represented is the one that performs best for this data set, *Comb 1.*

Table 2 Classification Accuracy for single classifier and ensemble of classifiers for *Localizer Experiment 1*

Accuracy (%)	*Number of features*		*Threshold in MI*	
	Best Fixed	*Estimation*	*Best Fixed*	*Estimation*
Single GNB	76.25	75.31	75.00	74.69
Ensemble of GNB	80.00	78.13	79.37	78.13
Single kNN	80.65	78.44	76.85	79.31
Ensemble of kNN	82.81	82.81	82.81	84.06

The ensembles of classifiers outperformed the single classifier for all the methods of estimation and fixed number presented in Table 2. The best performing classifier combination method is *Comb 1*; this method bases the support for a given class solely in the support given by the respective classifier. Suggesting that each stimuli is best discriminated by a single or conjunct of brain regions.

As in the previous section, only the results for the GNB classifier with *Threshold Estimation* as the feature selection method are represented in Fig. 10.

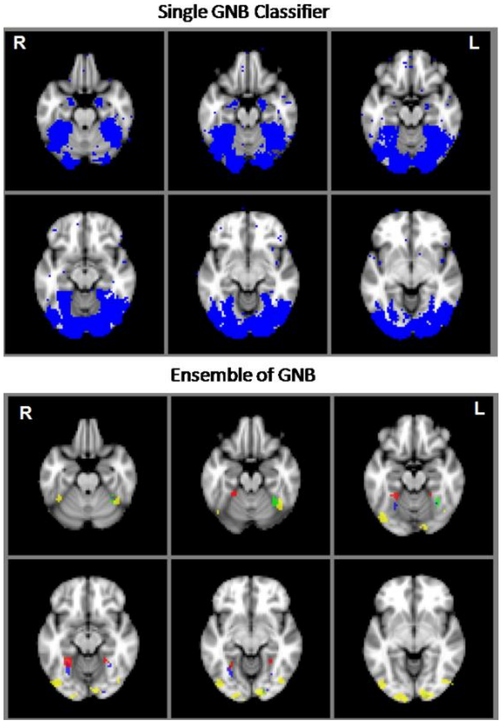


Figure 10 Voxels corresponding to the features selected for the *Localizer Experiment 1* by the *Threshold Estimation* method for GNB classifier

The voxels corresponding to the feature selected for the single classifier are localized in the expected activated areas. However besides the specific areas for each stimuli are also selected the neighbour voxels as consequence of the BOLD signal spatial properties. In what concerns the features selected for the ensemble of classifiers, the number of features is much smaller and the regions selected for each show little or even any overlap, defining compact regions associated with each stimuli. Both estimation methods performed satisfactorily with the *Threshold Estimation* method outperforming for both single classifier and ensemble of classifiers the best value for fixed MI threshold.

For this data set SVM was the best single classifier method with 91.56% of classification accuracy for the *Number Estimation* method and 93.44% for the *Threshold Estimation* method. These results were expected as consequence of the much more powerful formulation of this classifier

c) Localizer Experiment 2

The results for the *Localizer Experiment 2* classification accuracy for both single classifier and ensemble of classifiers using kNN and GNB are shown in Table 3. The classifier combination represented is the best performing, *Comb 1*.

Table 3 Classification Accuracy for single classifier and ensemble of classifiers for *Localizer Experiment 2*

Accuracy (%)	*Number of features*		*Threshold in MI*	
	Best Fixed	*Estimation*	*Best Fixed*	*Estimation*
Single GNB	70.83	68.09	71.53	71.53
Ensemble of GNB	75.89	71.53	75.69	73.61
Single kNN	74.81	66.67	75.00	71.53
Ensemble of kNN	77.08	75.00	82.25	79.17

As for the *Localizer Experiment 1* the ensemble of classifiers outperforms the single classifiers for all the methods and classifiers and the *Comb 1* is again the best performing classifier combination method. This concordance of methods performance is expected as the experience is the same. The features selected for this data set using the *Threshold Estimation* method of feature selection and the GNB classifier are shown in Fig. 11.

For the single classifier the features selected are much similar to the ones found for *Localizer Experiment 1*, although the increase of resolution for the ensemble of classifiers are chosen a small amount of feature when compared with the previous data set. This fact is counter intuitive as for a higher resolution a given region is defined by a larger number of features. However the results are supported by the study with a fixed number of features as the best accuracies were achieved with a smaller number of features.

The estimation methods have a poorer performance for this data set, in particular for the *Number Estimation* method. The differences between the estimation methods are related with the fact that *Threshold Estimation* method has better resolution in range of the small number of features, the interest zone for the ensemble of classifiers. The overall difference between *Localizer Experiment 1* and *Localizer Experiment 2*, can be explained by the difference in the number of examples available, inter-subject variability and SNR differences. Although the increase of magnetic field is associated with the increase of SNR the increase of resolution has the contrary effect.

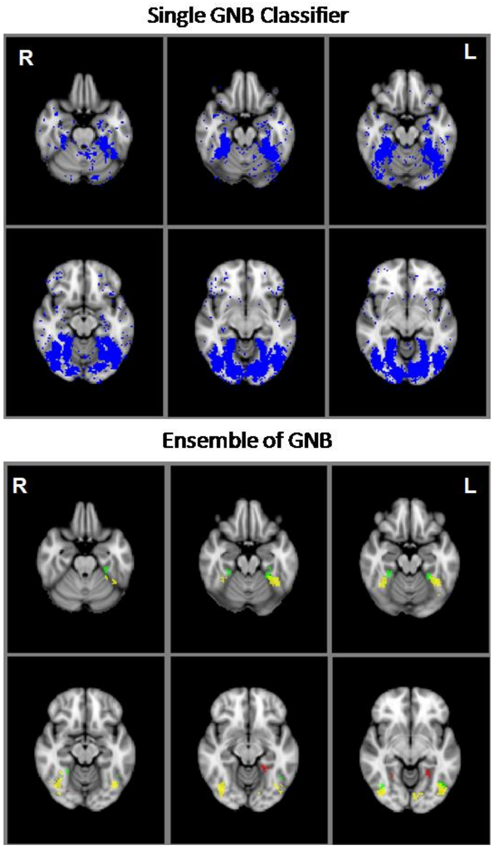


Figure 11 Voxels corresponding to the features selected for the *Localizer Experiment 2* by the *Threshold Estimation* method for GNB classifier

D. Multivariate Feature Selection Method

The results for the multivariate feature selection method for all the three data sets are shown on Table 4. The results for the best performing univariate single classifier data driven method are shown for comparison purposes.

Table 4 Classification accuracy for all data sets using the Multivariate feature selection method

Accuracy (%)		*Mapping Experiment*	*Localizer Experiment 1*	*Localizer Experiment 2*
GNB	Multivariate	97.85	75.63	71.53
	Univariate	98.63	75.21	71.53
kNN	Multivariate	98.24	79.06	73.61
	Univariate	99.41	79.38	71.53
SVM	Multivariate	99.02	90.31	81.94
	Univariate	99.41	93.44	82.64

The features selected for the three data sets for the GNB classifier are shown in Fig. 12.

The features selected for the multivariate feature selection method are localized in the expected regions with a compact and geometric form as consequence of the method.

Although the simplicity and the geometric constraints of the method the results are in overall as good as the best univariate single classifier data driven method even outperforming it in some cases. For SVM there are no improvements probably a consequence of the non utilization of this classifier as learning machine in the feature selection method.

IV. Conclusion

The main goal of this study was to detect cognitive states from a brain activation pattern; this goal was achieved, as the classification accuracy for all the methods was superior to the expected classification accuracy for the random classifier. Despite the inter-subject variability the classifiers are indeed learning and have predictive power. The features selected show good agreement with the expected activated zones. The classifier combination results suggest that an ensemble of classifiers can improve classification accuracy, while playing an important role in pattern localization and characterization. The automatic estimation method represents an attempt to reduce the requirement for prior knowledge of the problem with satisfactory results. The multivariate feature selection method proposed is very simple but is showed advantages in some contexts and thus could lead to a more accurate classification with further development.

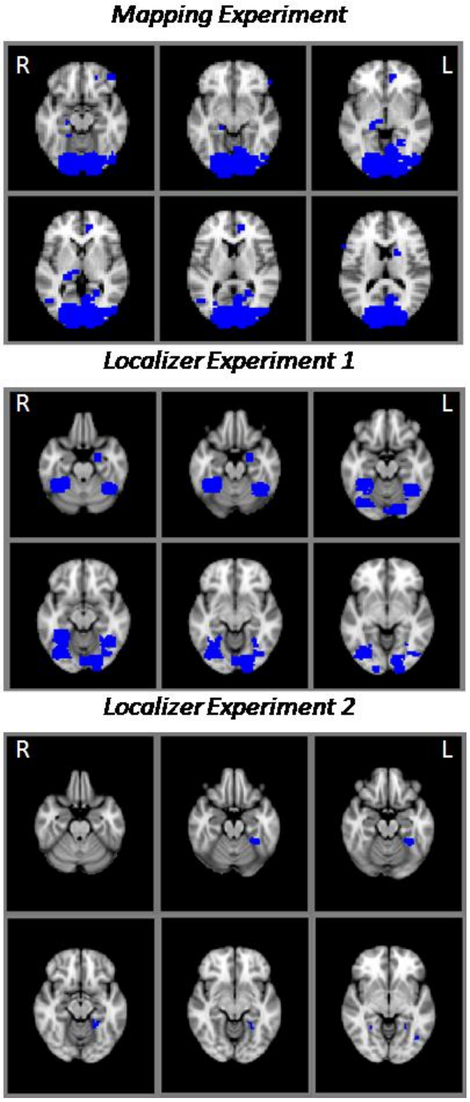


Figure 12 Voxels corresponding to the features selected for the data sets by the multivariate feature selection method for GNB

References

1. Jezzard, P., P. Matthews, and S. Smith, *Functional MRI: an introduction to methods*. 2001: Oxford University Press.
2. Friston, K., et al., *Statistical parametric maps in functional imaging: a general linear approach.* Human brain mapping, 1994. **2**(4): p. 189-210.
3. Mitchell, T., et al., *Learning to decode cognitive states from brain images.* Machine Learning, 2004. **57**(1): p. 145-175.
4. O'Toole, A., et al., *Theoretical, statistical, and practical perspectives on pattern-based classification approaches to the analysis of functional neuroimaging data.* Journal of cognitive neuroscience, 2007. **19**(11): p. 1735-1752.
5. Haynes, J. and G. Rees, *Decoding mental states from brain activity in humans.* Nature Reviews Neuroscience, 2006. **7**(7): p. 523-534.
6. Norman, K., et al., *Beyond mind-reading: multi-voxel pattern analysis of fMRI data.* Trends in cognitive sciences, 2006. **10**(9): p. 424-430.
7. Pereira, F., T. Mitchell, and M. Botvinick, *Machine learning classifiers and fMRI: a tutorial overview.* NeuroImage, 2009. **45**(1): p. S199-S209.
8. Grill-Spector, K. and R. Malach, *The human visual cortex.* Neuroscience, 2004. **27**(1): p. 649.
9. Haxby, J., et al., *Distributed and overlapping representations of faces and objects in ventral temporal cortex.* Science, 2001. **293**(5539): p. 2425.
10. Mitchell, T., *Machine Learning*. 1997: McGraw-Hill.
11. Zhang, L., et al., *Machine learning for clinical diagnosis from functional magnetic resonance imaging.* IEEE Internacional Conference Computer Vision and Pattern Recognition, 2005. **I**: p. 1211-1217.
12. Wang, X., R. Hutchinson, and T. Mitchell. *Training fMRI classifiers to detect cognitive states across multiple human subjects*. in *NIPS03*. 2003.
13. Hanson, S. and Y. Halchenko, *Brain Reading Using Full Brain Support Vector Machines for Object Recognition: There Is No Face" Identification Area.* Neural Computation, 2008. **20**(2): p. 486-503.
14. Guyon, I., et al., *Feature extraction: foundations and applications*. 2006: Springer Verlag.
15. Mourão-Miranda, J., et al., *Classifying brain states and determining the discriminating activation patterns: support vector machine on functional MRI data.* NeuroImage, 2005. **28**(4): p. 980-995.
16. Dietterich, T., *Ensemble methods in machine learning.* Multiple classifier systems, 2000: p. 1-15.
17. Kuncheva, L., *Combining pattern classifiers: methods and algorithms*. 2004: Wiley-Interscience.
18. Richiardi, J., et al. *Brain decoding of FMRI connectivity graphs using decision tree ensembles*. 2010: IEEE Press.
19. Kuncheva, L. and J. Rodríguez, *Classifier ensembles for fMRI data analysis: an experiment.* Magnetic resonance imaging, 2010. **28**(4): p. 583-593.
20. Cruz, P., J. Teixeira, and P. Figueiredo, *Reproducibility of a rapid visual brain mapping protocol*, in *Human Brain Mapping 2009*. 2009.
21. Saiote, C., et al., *Parametric fMRI correlates of multiple face orientation*, in *16th Annual Meeting of the Organization for Human Brain Mapping* 2010: Barcelona, Spain.
22. Jenkinson, M., et al., *Improved optimization for the robust and accurate linear registration and motion correction of brain images.* NeuroImage, 2002. **17**(2): p. 825-841.
23. Smith, S., *Fast robust automated brain extraction.* Human brain mapping, 2002. **17**(3): p. 143-155.
24. Lancaster, J., et al., *Bias between MNI and Talairach coordinates analyzed using the ICBM 152 brain template.* Human brain mapping, 2007. **28**(11): p. 1194-1205.
25. Chang, C. and C. Lin, *LIBSVM: a library for support vector machines.* 2001.
26. Harris, F., *On the use of windows for harmonic analysis with the discrete Fourier transform.* Proceedings of the IEEE, 1978. **66**(1): p. 51-83.
27. Ni, B. and J. Liu, *A hybrid filter/wrapper gene selection method for microarray classification*, in *IEEE Procedings of 2004 Internacional Conference on Machine Learning and Cybernetics*. 2005, IEEE. p. 2537-2542.