Nonlinear dimension reduction and activation detection for fMRI dataset

Xilin Shen
University of Colorado at Boulder
Boulder, CO, 80309
xshen@colorado.edu

François G. Meyer
Francois.Meyer@colorado.edu

Abstract

Functional magnetic resonance imaging (fMRI) has been established as a powerful method for brain mapping. Different physical phenomena contribute to the dynamical changes in the fMRI signal, the task-related hemodynamic responses, non-task-related physiological rhythms, machine and motion artifacts, etc. In this paper, we propose a new approach for fMRI data analysis. Each fMRI time series is viewed as a point in $\mathbb{R}^T$. We are interested in learning the organization of the points in high dimensions and extracting useful information for data classification. A nonlinear manifold learning technique is applied to obtain a low dimensional embedding of a dataset. The embedding differentiates time series with different temporal patterns. By assuming that the subset of activated time series forms a low dimensional structure, we partition the dataset and separate subsets of points with low dimensionality. The correspondence between low dimensional subsets and time series that contain task-related responses is verified and the activation maps are generated accordingly. The proposed approach is data-driven. It does not require a model for the hemodynamic response. We have conducted several experiments with synthetic and in-vivo datasets that demonstrate the performance of our approach.

1. Introduction

Instead of measuring the electrical events of the neuron population directly, BOLD (blood oxygen level-dependent) fMRI measures the hemodynamic and metabolic changes of the neural activities upon external stimulus [6]. During a neuron-imaging experiment, a three dimensional fMRI signal is acquired with a repetition time (TR) of around several seconds. A time series $\mathbf{x}_i = [x_i(1), x_i(2), \ldots, x_i(T)]^T$ is obtained for every voxel and it is indexed by its spatial position i. The goal of fMRI data analysis is to detect time series that contain the task-related hemodynamic responses.

The most popular method for fMRI data analysis is the Statistical Parametric Mapping (SPM) [12], which is based on a univariate regression model,

$$\mathbf{x}_i = \Gamma\beta + \varepsilon. \quad (1)$$

SPM employs a hypothesis driven approach to test if the time series contain a task-related response. It computes the t-statistic at each voxel and the statistical parametric map is formed by thresholding the p-value.

Exploratory methods are capable of identifying non-trivial components, which helps to understand the nature of the confounds and generate new hypothesis. These methods include multivariate methods, *e.g.*, principal component analysis (PCA) [18] and independent component analysis (ICA) [16, 15, 11] and clustering-based methods [1, 17, 10]. PCA and ICA are both non-inferential methods and they try to decompose fMRI time series into a linear combination of temporal patterns which explain the variability in the fMRI signal. These methods do not require a model for the hemodynamic response function. However, the orthogonality or the statistical independence imposed by PCA or ICA does not have a direct physiological interpretation.

Instead of applying linear decomposition to time series as is being done in SPM, PCA or ICA, we propose to analyze the geometry of fMRI datasets. Each time series is viewed as a point in $\mathbb{R}^T$, where T is the number of discretized time indices. The geometry of an fMRI dataset is defined in terms of how data points are organized with respect to other points in the high dimensional space. Note that the word "geometry" we use here has nothing to do with the spatial positions of the voxels. For fMRI data analysis, we are particularly interested in the organization of the activated time series and its relation to the geometry of the entire dataset. To obtain a partition of the dataset where activated time series are separated from the background, a good knowledge of the organization of the dataset can be very useful. The contribution of the paper is that it proposes to use nonlinear manifold learning algorithms to learn the geometry of the high dimensional fMRI dataset. The low dimensional embedding constructed as a result of the learning algorithm differentiates subsets of time series with different temporal patterns. It is also assumed in our work that the subset of activated time series has much smaller dimension-

Proceedings of the 2006 Conference on Computer Vision and Pattern Recognition Workshop (CVPRW'06)
0-7695-2646-2/06 $20.00 © 2006 IEEE

ality than the dimensionality T of the ambient space. Therefore we separate subsets of points with low dimensionality by partitioning in the embedding space. The activated time series are then identified and the activation maps are generated accordingly. The paper is organized as follows: in section II, we give an overall description of our approach. A brief overview of manifold learning and nonlinear dimension reduction techniques [20, 13, 8, 19, 2] is given in section III, and we explain the diffusion map algorithm used in this work. Section IV describes the experiment and shows the result for both synthetic and in-vivo fMRI datasets. We discuss some relevant issues in section V and conclude in section VI.

2. Proposed Method

Various physical sources contribute to the dynamic changes in fMRI time series. We call signals that are associated with well-defined physical processes the *source signals*. They include task-related hemodynamic response, non-task-related physiological rhythms like breathing and heart beating, motion or machine artifacts. In general, a small proportion of time series in a fMRI dataset contain either one of *source signals* described above. Whereas the majority of time series are generated as a result of thermal noise. We are interested in the geometry formed by the different time series in $\mathbb{R}^T$, and our goal is to partition the dataset based on the knowledge of the organization of the dataset. However, analysis in high dimensions is known to be hard. Thus traditional clustering techniques apply linear dimension reduction to datasets before the partitioning. Linear methods obtain the low dimensional embedding by projecting each data point onto a series of basis functions (fixed or adaptively chosen). The linear mapping may fail to preserve local proximity, thus obscures the geometry of the dataset in the low dimensional embedding. Especially in the presence of nonlinearity, a possibility that should not be excluded for fMRI datasets, clustering based on linear dimension reduction could be problematic.

In the proposed approach, we employ nonlinear techniques to reduce the dimensionality of the datasets. Nonlinear techniques learn the organization of the dataset by integrating local proximity information, so the low dimensional embedding preserves local structures. Furthermore, we make an assumption that time series that contain a single *source signal* are distributed in a way that they form a low dimensional structure. In particular, the set of time series that contain task-related hemodynamic responses has an intrinsic dimensionality much smaller than the dimensionality T of the ambient space. We also assume that the set of time series from background voxels has a much larger dimensionality. In the high dimensional space, low dimensional structures corresponding to the different *source signals* are well separated from each other and they are separated from the subset of background time series as well. Therefore, a desirable low dimensional embedding should not only exhibit the existence of such low dimensional structures, but also represent each of them in a distinctive way that makes data partitioning straightforward. With such an embedding, the subset of activated time series could be easily identified and separated from the rest of the data. The proposed approach includes the following three steps:

- **Dimension reduction using nonlinear techniques**: we apply the diffusion map algorithm to fMRI datasets and obtain a low dimensional embedding.
- **Separation of subsets with low dimensionality in the embedding space**: we discover coherent structures that are almost one dimensional in the embedding. By clustering in the embedding space, we separate the low dimensional subsets from the rest of the dataset.
- **Identifying the subsets of activated time series and generating the activation maps**: we examine temporal patterns associated with each low dimensional subset and identify the one subset that corresponds to time series that contain task-related responses.

Now we give a justification for the low dimensionality assumption of the activated time series. Blood oxygenation level dependent (BOLD) fMRI measures the hemodynamic response to brain activation. The response mainly involves a local increase of blood flow. The increase may vary in the onset, the magnitude and the duration of the response. Simple parametric models have been successful in mathematical modelling of the hemodynamic response. The canonical hemodynamic response function (HRF) (defined in (11)) used by SPM is an example of such models. If we construct the set of hemodynamic responses from all possible values of the parameters, it would form a manifold in $\mathbb{R}^T$. The intrinsic dimensionality of the manifold is determined by the number of independent parameters, and it is much smaller than the dimensionality of the ambient space $\mathbb{R}^T$. In reality, it is inevitable that the time series are corrupted by noise. Well, it is still reasonable to believe the low dimensionality assumption holds for the activated time series. The assumption is also consistent with the empirical findings by PCA and ICA, where a small number of temporal patterns are sufficient to describe the variations of activated time series.

3. Manifold learning and Nonlinear dimension reduction

Nonlinear manifold learning algorithms, including isomap [20], locally linear embedding (LLE) [19], Laplacian eigenmap [2], and diffusion map [8, 13], have in common the idea of integrating local proximity information to form a global representation of the manifold. The general procedure starts by representing the dataset in the form of a graph. Spectral analysis of certain matrices associated

with the graph gives the embedding coordinates in low dimensions. Isomap approximates geodesic distance between data points and applies the classical multidimensional scaling (MDS) [4] to the distance matrix. LLE assumes local linearity and reconstruct the dataset according to the optimal local weights. Laplacian Eigenmap works with the graph Laplacian matrix. The diffusion map is very similar computational-wise to Laplacian Eigenmap. However it takes a probabilistic perspective. It defines a random walk on a dataset and characterizes the distance between pairs of points by their transition probability. We will discuss in the following the diffusion map algorithm and some details regarding implementation. More information regarding the algorithm could be found in [8, 13].

3.1. Diffusion Map

The first step of the diffusion map algorithm is representing the dataset $X = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ by a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$. Each time series $\mathbf{x}_i$ is regarded as one vertex in the graph and each edge in $\mathscr{E}$ is associated with a weight function $w(\mathbf{x}_i, \mathbf{x}_j)$. The weighted graph $\mathscr{G}$ is fully described by the weight matrix W, where each entry of W is denoted as W_{ij}.

The diffusion map algorithm defines a Markov random walk on the graph $\mathscr{G}$. The idea is that the propagation of a random walk depends on the graph structure. If there are a lot of short paths between two vertices $\mathbf{x}_i$ and $\mathbf{x}_j$, it becomes relatively easy to reach $\mathbf{x}_i$ from $\mathbf{x}_j$ and vice versa. Thus these two points should be considered close. The goal of the diffusion map algorithm is to define a distance metric that reflects such connectivity. The distance metric is called the *diffusion distance* and it is based on the transition probability between pairs of points. The transition probability of going from point $\mathbf{x}_i$ to $\mathbf{x}_j$ in one step is defined as:

$$p(\mathbf{x}_j|\mathbf{x}_i) = \frac{W_{ij}}{\sum_k W_{ik}}. \tag{2}$$

We denote P as the transition matrix of the random walk with its entry $P_{ij} = p(\mathbf{x}_j|\mathbf{x}_i)$. To overcome the inconvenient fact that P is non-symmetric, we instead consider the symmetric conjugate of P,

$$A = R^{\frac{1}{2}} P R^{-\frac{1}{2}} = R^{-\frac{1}{2}} W R^{-\frac{1}{2}}. \tag{3}$$

We call A the diffusion matrix of the random walk and R is a diagonal matrix with entry $R_{ii} = \sum_j W_{ij}$. The diffusion distance is defined as:

$$(d_{ij}^{2m}) = \|A_{i\cdot}^m - A_{j\cdot}^m\|, \tag{4}$$

where A^m is the mth power of A. The diffusion distance d_{ij}^{2m} is a sum over all paths of length less than or equal to $2m$ between $\mathbf{x}_i$ and $\mathbf{x}_j$. The diffusion distance has the property that it decreases when the number of paths between $\mathbf{x}_i$ and $\mathbf{x}_j$ increases. So it is a good measure of connectivity. Moreover it is robust to noise [14].

To obtain the set of embedding coordinates that convert the diffusion distance into Euclidean distance, we apply the spectral analysis to the diffusion matrix A. It is known that A has a set of right eigenvectors $\{\phi_k\}$ and a set of eigenvalues $\{\lambda_k, 1 = \lambda_0 \geq \lambda_1 \geq \lambda_2 \geq \ldots \geq 0\}$,

$$A\phi_k = \lambda_k \phi_k. \tag{5}$$

A result of spectral graph theory [7] shows that each entry of the diffusion matrix A^m can be expressed by its eigenvectors and eigenvalues,

$$A_{ij}^m = \sum_k \lambda_k^m \phi_k(i)\phi_k(j). \tag{6}$$

The K-dimensional diffusion map is defined as,

$$\Phi_K^m : \mathbf{x}_i \mapsto \begin{pmatrix} \lambda_0^m \phi_0(\mathbf{x}_i) \\ \lambda_1^m \phi_1(\mathbf{x}_i) \\ \lambda_2^m \phi_2(\mathbf{x}_i) \\ \cdot \\ \cdot \\ \lambda_K^m \phi_K(\mathbf{x}_i) \end{pmatrix} \tag{7}$$

Therefore the K-dimensional diffusion map approximates the diffusion distance by Euclidean distance,

$$(d_{ij}^{2m})^2 \approx \sum_k^K (\lambda_k^m \phi_k(i) - \lambda_k^m \phi_k(j))^2 = \|\Phi_K^m(\mathbf{x}_i) - \Phi_K^m(\mathbf{x}_j)\|^2 \tag{8}$$

The K-dimensional embedding coordinates for point $\mathbf{x}_i$ are $[\lambda_1^m \phi_1(\mathbf{x}_i), \lambda_2^m \phi_2(\mathbf{x}_i), \ldots, \lambda_K^m \phi_K(\mathbf{x}_i)]$. Dimension reduction is achieved by choosing K smaller than the dimensionality T of the ambient space. The diffusion map algorithm is summarized in the following table.

Table 1. Diffusion map algorithm

step 1:	construct a graph $\mathscr{G} = (\mathscr{V}, \mathscr{E})$ from the dataset;
step 2:	assign a weight function to edges defined in E and obtain a symmetric weight matrix W;
step 3:	compute the diffusion matrix A defined by (3);
step 4:	solve the eigen problem $A\phi = \lambda\phi$;
step 5:	compute the diffusion map defined by (7).

3.2. Details about the implementation

3.2.1 Graph construction and weight assignment

To represent the dataset in the form of a graph $\mathscr{G}$, we need to specify the vertex set $\mathscr{V}$ and the edge set $\mathscr{E}$. We consider each time series $\mathbf{x}_i$ as a vertex and find k nearest vertices to $\mathbf{x}_i$ and put an edge between each one of these vertices and $\mathbf{x}_i$. The k nearest neighbour method guarantees to give a connected graph.

In literature on diffusion map [13] and Laplacian eigenmap [2], the Gaussian kernel is chosen to be the weight function. From a theoretical point of view, a properly normalized Gaussian kernel approximates the heat diffusion on

a Riemannian manifold [13]. By studying the way the heat diffuses on the manifold, one learns about the geometry of the manifold. In this work, there is no explicit Riemannian manifold, we nevertheless favour the use of the Gaussian kernel based on the following reasons. First, the Gaussian kernel is rotation invariant, so it does not have bias toward any direction for the random walk. Besides, the kernel has an exponential decay. The neighbouring nodes in the set $\mathscr{N}(\mathbf{x}_i)$ of a node $\mathbf{x}_i$ may be at various different distances from $\mathbf{x}_i$. The exponential decay ensures that transition between closer neighbours will take place with a much higher probability. So it reinforces the local geometry. The weights are computed as

$$W_{ij} = \begin{cases} \exp\left[-\left(\frac{d_{ij}}{\sigma}\right)^2\right] & \text{if } \mathbf{x}_i \text{ are connected to } \mathbf{x}_j \ , \\ 0 & \text{if } \mathbf{x}_i \text{ are not connected to } \mathbf{x}_j \ , \end{cases} \tag{9}$$

where σ is the scale parameter, d_{ij} is the Euclidean distance between time series $\mathbf{x}_i$ and $\mathbf{x}_j$.

Note that the position (spatial coordinates) of the voxels from which the time series are extracted is not used in the weight assignment. We know that spatial information is helpful in the sense that in most cases, truly activated voxels tend to be spatially clustered and we can exploit this property in the detection of activation. However, because of the complexity of the cortical surface, neighbouring voxels may belong to different functional units, thus produce entirely different responses. So putting edges between time series from spatial neighbours is not always appropriate.

3.2.2 Determining the dimensionality

The diffusion map algorithm yields for each vertex in the graph a set of embedding coordinates based on eigendecomposition. However, the dimensionality itself is not specified by the algorithm and needs to be estimated. In some applications of the diffusion map algorithm, the estimation of dimensionality is based on a precision value δ given in advance. One looks for K, such that $|\lambda_i|^m \geq \delta|\lambda_0|^m, i = 0, 1, \ldots, K$. The amount of residue variance is controlled in this way. In our application, we do not intend to control the residue variance of the whole dataset. What we are really interested in is whether the embedding reveals low dimensional features. In general, the first few coordinates are the effective coordinates that describe the variance within the low dimensional subsets. The subsequent coordinates contributes to model the variations in the high dimensional subsets of the dataset. In the experiments we usually choose the first three to ten coordinates.

4. Experiments and results

For each dataset, time series from regions of interest are extracted for the analysis. One would like to include all time series inside the brain for the data analysis. However, due to computational concern, we work with a reasonable number of time series in the experiments. Following the steps listed in Table 1, we apply the diffusion map algorithm to each dataset and obtain the embedding. We then use the K-Means clustering algorithm [3] to partition the dataset in the embedding space. Time series associated with each subsets of low dimensionality are examined and subsets of activated time series are identified.

We evaluate our approach according to two criteria. First, we compare the nonlinear dimension reduction algorithm with PCA. The criterion is whether the embeddings exhibit low dimensional features and whether they correspond to the set of time series that contain *source signals*. Additionally, we compare the activation detection result with the one obtained by using the General Linear Model (GLM). Two datasets are selected for the analysis. The first dataset is a synthetic dataset. It is used for validation and provides a statistical comparison with the GLM in terms of type I and type II errors. The other dataset is an in-vivo fMRI dataset with an event-related design.

4.1. Synthetic data

Since validation is difficult in the case of real fMRI data, a synthetic data slice is constructed to evaluate the performance of our proposed approach. There are altogether 1067 voxels from the slice, among which 97 are assigned to be the activated. The percentage of activation is approximately 9%. The true activation map of the slice is shown in Fig. 1. The white circular region in the center of Fig. 1 is the activation focus. Voxels in the gray area are considered to be inside the brain, but remain non-activated. The black region is the region outside the brain.

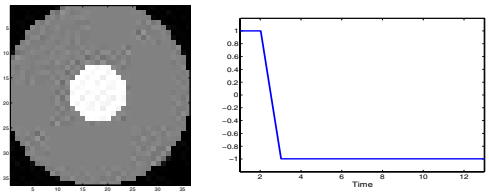

Figure 1. Left: true activation map of the synthetic data slice. White: region of activation; Gray: region of non-activation; Black: region outside the brain. Right: experiment paradigm.

The synthetic data is generated by superimposing simulated activated time series upon real fMRI data. fMRI time series are selected from an in vivo dataset given in [17]. We extract time series from regions where no activation is supposed to be elicited and eliminate those exhibiting large variance. The noise in the synthetic dataset is not white but has temporal autocorrelation [22]. No obvious confounding time series are included in the synthetic dataset. For voxels in the non-activated region (gray area), we simply take

the non-activated time series from the real fMRI data. For each voxel in the activated region (white area), the associated time series is generated by,

$$f(t) = h(t) \star g(t) + n(t), \tag{10}$$

where $g(t)$ is the experiment paradigm shown in Fig. 1, $h(t)$ is the HRF defined by (11), and $n(t)$ is the background time series from real fMRI data. Examples of non-activated and activated time series are also shown in Fig. 2.

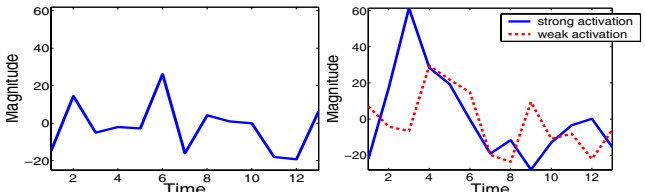

Figure 2. Left: an example of non-activated time series. Right: two examples of activated time series, one with stronger activation and one with weaker activation.

The canonical hemodynamic response function [21] is defined as,

$$h(t)=\left(\frac{t}{d_1}\right)^{a_1}\exp\left(-\frac{t-d_1}{b_1}\right)-c\left(\frac{t}{d_2}\right)^{a_2}\exp\left(-\frac{t-d_2}{b_2}\right), \tag{11}$$

By varying the dispersion parameter b_1, we obtain a series of hemodynamic responses with different peak delays and shapes. In addition, we scale the hemodynamic response by α to model the different levels of activation strength. For the data generation, we take b_1 to be uniformly distributed between 0.8 and 1.2, and α to be uniformly distributed between 5 and 10. The two random variables are independent. The values for the rest of the parameters are $a_1 = 6$, $c = 0.35$, $a_2 = 12$, $b_2 = 0.9$. We generate 20 independent synthetic datasets according to the design.

The 2-dimensional embedding of the synthetic dataset given by the diffusion map algorithm is shown in Fig. 3. Since we know the ground truth of the activation map, different symbols are used to distinguish points from activated region and points from non-activated region in Fig. 3.

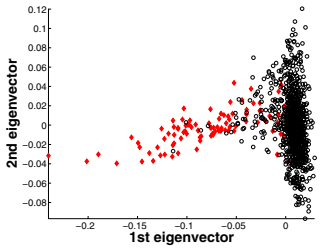

Figure 3. 2D embedding given by the diffusion map algorithm. Red diamonds are points from the activated region. Black filled circles are points from the non-activated region.

The embedding given by the diffusion map algorithm shows points from the activated region (red diamonds) are distributed along a thin strip that extends outward from the main cluster of points. This low dimensional structure is compact and easy to identify. We also construct the 2-dimensional embedding by PCA, see Fig. 4. However, we do not see any low dimensional structures. Points from the subset of activated time series are distributed as outliers to the main point cloud.

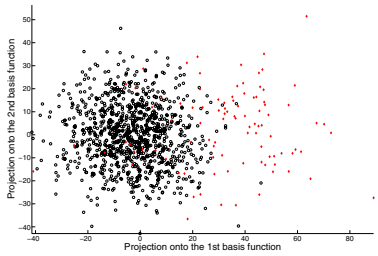

Figure 4. 2D embedding given by PCA. Red diamonds are points from the activated region. Black filled circles are points from the non-activated region.

K-Means clustering in the embedding space partitions the data into 2 clusters. The number of cluster is given to the algorithm. The partition result is shown in Fig. 5. One cluster corresponds to the activated time series, and the other cluster corresponds to time series from the non-activated region. The corresponding activation map is shown in Fig. 6.

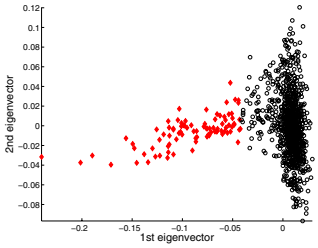

Figure 5. Partition result: points of red diamond belong to cluster I, points of black circles belong to cluster II.

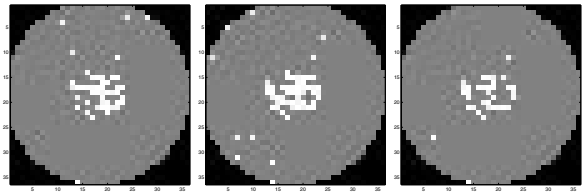

Figure 6. Left: activation map obtained by K-Means clustering based on the embedding; Middle: activation map obtained by thresholding the p-value at 0.005. Right: activation map obtained by thresholding the p-value at 0.001.

We now compare the above activation map with the one obtained by using the General Linear Model. We have the exact knowledge of the task-related hemodynamic response associated with each voxel from the activated region, and we use them for the regression. For time series from non-activated voxels, we use $h(t)$ with $b_1 = 1$ and $\alpha = 1$ to generate the regressor. Two activation maps obtained by thresholding the p-value at 0.005 and 0.001 respectively are shown in Fig. 6. The statistics of type I error (false alarms)

and type II error (missing activation) are summarized in Table (2) and Table (3) . Because the synthetic dataset contains activated time series at different strength level, we separate them into five subgroups in terms of the values of α and the statistics are obtained accordingly.

Table 2. The average number of type I error calculated from 20 synthetic datasets.

Method	Type I error
K-Means clustering based on the embedding	18.5
p-value thresholding at 0.001	4.65
p-value thresholding at 0.005	11.75

Table 3. The average percentage of activated voxels that have been classifi ed as non-activated by mistake.

α value	K-Means clustering based on the embedding	p-value thresholding at 0.005	p-value thresholding at 0.001
5 to 6	0.4710	0.5223	0.7567
6 to 7	0.3632	0.3806	0.6119
7 to 8	0.2569	0.2265	0.4779
8 to 9	0.1508	0.1201	0.3324
9 to 10	0.1108	0.0757	0.1973

From Table (2) and Table (3), we see that our approach tends to create more false alarms then the linear model. However, in terms of type II error, the simple clustering algorithm outperforms the result by thresholding the p-value at 0.001 and has comparably good performance as thresholding the p-value at 0.005. Note that the linear model has access to the "oracle". If the non-activated time series are white Gaussian, the linear model would yield the optimal result. In this example, the non-activated time series are extracted from in vivo fMRI dataset, so they are not exactly realizations of a white Gaussian random process.

4.2. In vivo data: event-related dataset

Buckner et al.[5] used fMRI to study the age-related changes in functional anatomy. Nondemented adults (both young and old) and demented older adults were recruited for the experiment. The subjects were instructed to press a key with their right index finger upon the visual stimulus onset. Functional images were collected using a Siemens 1.5-T Vision System with an asymmetric spin-echo sequence sensitive to BOLD contrast (volume TR = 2.68 sec, 3.75×3.75mm in-plane resolution; T2* evolution time = 50 msec, alpha = 90°). Sixteen contiguous axial slices were acquired. For each slice, 128 sequential images were obtained consisting of one run of functional imaging. The image resolution is 64×64. For every 8 images, the subjects were presented with one of the two trial conditions. The so called one-trial condition involves one isolate stimulus. The two-trial condition has two consecutive stimuli with an inter-stimulus interval of 5.36 sec. The one-trial conditions and two-trial conditions were mixed in a pseudo random fashion. There were 15 trials per run.

We use one dataset from a single subject consisting of a single run of experiment. The subject is an 80 year old female with mild dementia. We are also given the corresponding ordering of the conditions of the trials. For the data analysis, the first and last four images are discarded. So the total length of time series is 120. Each run of experiment consists of 15 trials, and each trial consists of 8 temporal samples. According to the ordering information, 8 of the 15 trials are of one-trial condition and the other 7 are of two-trial condition. Time series from one-trial and two-trial conditions are averaged separately. Therefore, each voxel is associated with two averaged time series. The linear trend is removed from all averaged time series.

The results published in [5] show activation in the visual cortex, motor cortex, and cerebellum. We focus our analysis to a volume extracted from the posterior region of the brain which encompasses the visual cortex. The volume includes 4 axial slices, with the original index 7, 8, 9 and 10. The region included within each slice is shown in Fig. 7.

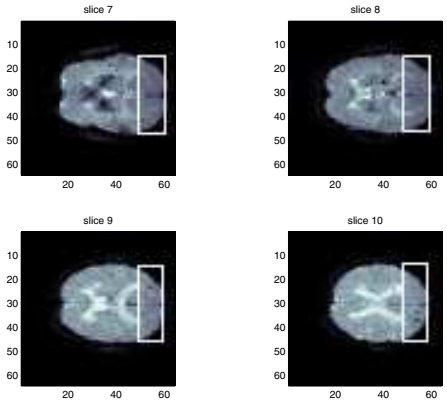

Figure 7. Intracranial voxels inside the white frame are selected for the data analysis.

The 3D embeddings given by the diffusion map algorithm is shown in Fig. 8. The embedding shows two distinctive "branches" forming a "V" shape. Both "branches" are nearly one dimensional. We are interested in learning the kind of time series that are associated with each of the two "branches" and see if one of them is related to the task-related responses. The partition result is superimposed on the embedding.

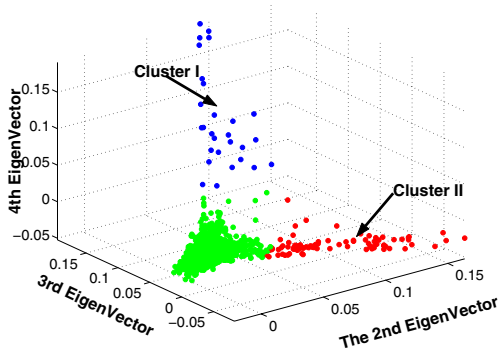

Figure 8. 3D embedding given by the diffusion map algorithm. The result of a three-cluster partitioning: blue points belong to cluster I, red points belong to cluster II, green points belong to background cluster.

Representative time series from cluster I (red) and cluster II (blue) defined in Fig. 8 are shown in Fig. 9. The corresponding activation maps are shown in Fig.10. The time series from cluster I has an abrupt dip at $t = 7$. The corresponding voxels are located along the border of the brain. Judging from these facts, it is most likely that time series from cluster I are created by motion artifact. However, the time series from cluster II have clearly the shape of a hemodynamic response, and the corresponding voxels are located in the region of visual cortex. Therefore, data points in cluster II are related to activation in the visual cortex.

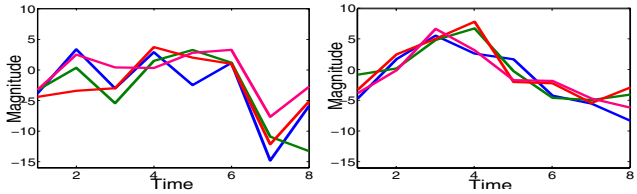

Figure 9. Left: representative time series from the red cluster in Fig. 8. Right: representative time series from the blue cluster.

We also obtain the activation map using the linear regression model. The averaged time series from two-trial conditions are used for the regression analysis. We use the same model for the hemodynamic response function as was used in [9],

$$h(t) = ((t-\delta)/\tau)^2 \, \mathbf{e}^{\frac{(t-\delta)}{\tau}}, \qquad (12)$$

where $\delta = 2.5$, $\tau = 1.5$. We threshold the p-value at 0.005, and the activation map is shown in Fig.11. We conclude that the two activation maps are consistent with each other.

5. Discussion

In our approach, the K-Means [3] clustering algorithm is used for partitioning the data in the embedding space. The K-Means algorithm does not estimate the number of clusters by itself. However, because of the existence of various *source signals* other than the task-related hemodynamic response, it is impractical to specify the number of clusters

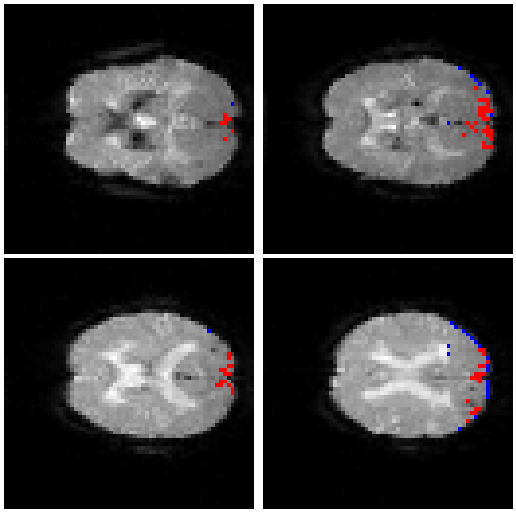

Figure 10. Activation maps corresponding to the partition result shown in Fig. 8. Voxels belonging to cluster I are highlighted in blue and voxels belonging to cluster II are highlighted in red.

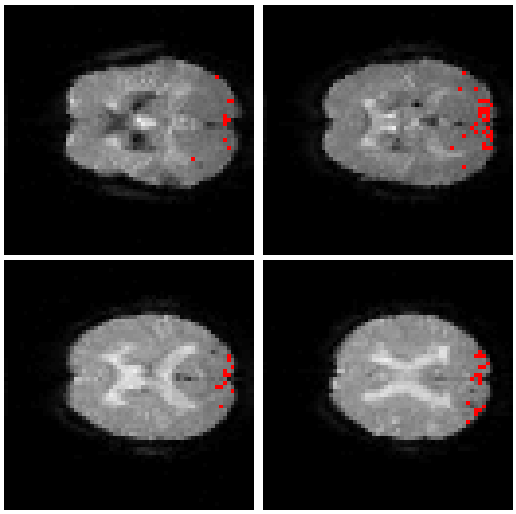

Figure 11. Activation maps using the linear regression model. The map is obtained by thresholding the p-value by 0.005. Activated voxels are highlighted using the red color.

in advance. In the experiment, the estimation of the number of clusters involves human interaction. We first look at the low dimensional embedding given by the diffusion map algorithm and check if there is any low dimensional structures. Based on our observation, we set the number of clusters to be the number of coherent low dimensional structures plus one (which represents the subset of background time series). In most cases, the low dimensional structures are evident and distinctive, so the estimation is rather easy. In certain cases when there is ambiguity in determining the number of clusters, we would try a few choices and choose the best one according to the criteria of selection: 1) the clustering should respect the coherence of the low dimen-

sional structures, we do not want to divide one low dimensional structure into several clusters; 2) the associated time series within each cluster should exhibit consistency to a certain degree. The estimation of number of clusters is done in a heuristic way and it can be improved in the further.

6. Conclusion

We have described in this paper a new approach for analyzing fMRI datasets. We view all time series as points in high dimensional space and assume that the time series from activated voxels have an intrinsic low dimensionality. Our results have shown that the embedding given by the diffusion map algorithm is capable of capturing the low dimensional features and clustering on the embedding space effectively separates the group of activated time series from other groups of time series. Result from the synthetic dataset indicates that our approach is at least as sensitive as the General Linear Model. However, since our approach is purely data-driven, it makes no assumption about the hemodynamic response function. Besides discovering task-related activation, it also finds other groups of time series generated from well defined sources. These time series could be related to functional responses which can be hardly anticipated before the experiment. The approach appears to be a powerful tool for exploratory analysis of fMRI data.

Acknowledgements

The Institute for Pure and Applied Mathematics (IPAM) at UCLA supported this work. The authors are extremely grateful to Dr. Mauro Maggioni and Dr. Stéphane Lafon for discussion of the theory and implementation of diffusion map algorithm and to Dr. Ivo Dinov for providing us with one of the in-vivo fMRI dataset.

References

[1] A. Baune, F. T. Sommer, et al. Dynamical cluster analysis of cortical fmri activation. *NeuroImage*, 9:477–489, 1999. 1

[2] M. Belkin and P. Niyogi. Laplacian eigenmaps for dimensionality reduction and data representation. *Neural Computation*, 15(6):1373–1396, 2003. 2, 3

[3] C. M. Bishop. *Neural Networks for Pattern Rgcognition*. Oxford University Press, 1995. 4, 7

[4] I. Borg and P. Groenen. *Modern Multidimensional Scaling: Theory and Applications*. Springer, 1997. 3

[5] R. L. Buckner et al. Functional brain imaging of young, nondemented, and demented older adults. *Journal of Cognitive Neuroscience*, 12:Supplement 2:24–34, 2000. 6

[6] W. Chen and S. Ogawa. Principles of bold functional mri. In C. T. W. Moonen and P. A. Bandettini, editors, *Functional MRI*. Springer-Verlag, Berlin Heidelberg, 1999. 1

[7] F. Chung. *Spectral Graph Theory*. CBNS-AMS, 1997. 3

[8] R. R. Coifman and S. Lafon. Diffusion maps. *Applied and Computational Harmonic Analysis*, submitted for publication. 2, 3

[9] A. M. Dale and R. L. Buckner. Selective averaging of rapidly presented individual trials using fmri. *Human Brain Mapping*, 5:329–340, 1997. 7

[10] P. Filzmoser, R. Baumgartner, and E. Moser. A hierarchical clustering method for analysis functional mr images. *Magnetic Resonance Imaging*, 17:817–826, 1999. 1

[11] K. J. Friston. Modes or models: a critique on independent component analysis for fmri. *Trends in Cognitive Sciences*, 2(10):373–375, 1998. 1

[12] K. J. Friston, J. Ashburner, et al. *SPM 97 course notes*. Wellcome Department of Cognitive Neurology, Univeristy College London, 1997. 1

[13] S. Lafon. *Diffusion maps and geometric harmonics*. PhD thesis, Yale University, New Haven, 2004. 2, 3, 4

[14] S. Lafon and A. Lee. Diffusion maps and coarse-graining: A unifi ed framework for dimensionality reduction, graph partitioning and data set parameterization. *IEEE transactions on Pattern Analysis and Machine Intelligence*, submitted for publication. 3

[15] M. J. McKeown, L. K. Hansen, and T. J. Sejnowski. Independent component analysis of functional mri: what is signal and what is noise? *Current Opinion in Neurobiology*, 13:620–629, 2003. 1

[16] M. J. McKeown, S. Makeig, et al. Analysis of fmri data by blind separaion into independent spatial components. *Human Brain Mapping*, 6:160–188, 1998. 1

[17] F. G. Meyer and J. Chinrungrueng. Local clustering of fmri time series in the frequency domain. *Medical Image Analysis*, 9(1):51–68, 2005. 1, 4

[18] K. M. Petersson et al. Statistical limitations in functional neuronimaging i. non-inferential methods and statistical models. *Phil. Trans. R. Soc. Lond. B*, 354:1239–1260, 1999. 1

[19] S. T. Roweis and L. K. Saul. Nonlinear dimensionality reduction by locally linear embedding. *Science*, 290:2323–2326, 2000. 2

[20] J. B. Tenenbaum, V. D. Silva, and J. C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290:2319–2322, 2000. 2

[21] K. J. Worsley et al. A general statistical analysis for fmri data. *NeuroImage*, 15:1–15, 2002. 5

[22] E. Zarahn, G. K. Aguirre, and M. D'Esposito. Empirical analyses of bold fmri statistics: I spatially unsmoothed data collected under null-hypothesis conditions. *NeuroImage*, 5:179–197, 1994. 4