8-1-2010

Numerical methods for fMRI data analysis

Geethmala Sridaran
Dartmouth College

Numerical methods for fMRI data analysis

Dartmouth Computer Science Technical Report TR2010-676

A Thesis

Submitted to the Faculty

in partial fulfillment of the requirements for the

degree of

Master of Science

in

Computer Science

by

Geethmala Sridaran

DARTMOUTH COLLEGE

Hanover, New Hampshire

August 2010

Examining Committee:

(chair) Richard Granger

Devin Balkcom

Peter Winkler

Brian W. Pogue, Ph.D.

Dean of Graduate Studies

Abstract

Brain imaging data are increasingly analyzed via a range of machine-learning methods. In this thesis, we discuss three specific contributions to the field of neuroimaging analysis methods:

1. To apply a recently-developed technique for identifying and viewing similarity structure in neuroimaging data, in which candidate representational structures are ranked (Kemp et al. [34]);

2. Provide side-by-side analyses of neuroimaging data by a typical non-hierarchical (SVM) versus hierarchical (Decision Tree) machine-learning classification methods; and

3. To develop a novel programming environment for PyMVPA, a current popular analysis toolbox, such that users will be able to type a small number of packaged commands to carry out a range of standard analyses.

We carried out our analysis with an fMRI data set generated using auditory stimuli. "Tree" and "Ring" were the best-voted structural representations we obtained by applying the Kemp et al. algorithm to our data. Machine-learning classification resulted in accuracy values that were similar for both decision tree and SVM algorithms. Coding for different sound categories primarily occurred in the temporal lobes of the brain. We discovered a few non-temporal regions of the brain coding for these auditory sounds as well.

Acknowledgments

I am grateful to my advisor Prof. Richard Granger, for his support and guidance throughout this project. He has been an amazing mentor, and I am indebted to him for his constant encouragement.

I am also thankful to my committee members Prof. Devin Balkcom and Prof. Peter Winkler for their valuable suggestions to this project.

I would also like to thank Prof. James V. Haxby, Prof. George Wolford, Yune-Sang Lee, Yaroslav O. Halchenko, Michael Hanke, Ashok Chandrashekar, James M. Hughes, Amy L. Palmer, Melissa M. Rundle, Andrew Connolly, and Rajeev Raizada for their guidance.

Lastly, I would like to thank my parents for constantly supporting my goals and ambitions throughout my education and career.

Parts of various chapters in this thesis have been reproduced from my thesis proposal document.

Contents

Appendix

List of Figures

Chapter 1

Introduction

Neuroimaging data can be analyzed using different machine-learning classifiers and numerical methods for extracting interesting information from the data. In this paper, we elucidate a few methods for fMRI data analysis with an illustration.

Pereira et al. [55] have explained how to apply machine-learning classifiers to fMRI data. In fMRI analysis, different voxels can be features, a column of voxels is an example, and a data set consists of groups of examples stacked on top of each other, as shown in Figure 1.1. A classifier is trained on various examples and tries to predict the labels. The data set is divided into training and test sets. The classifier is trained on the training data set, and the trained classifier is used to predict the labels for the test set as shown in Figure 1.2.[1] The measure of how good a classifier is performing is computed using a fraction of correctly predicted labels, and it is known as *accuracy*. Neuroimaging data usually has a lot of features and fewer examples. Hence it is desirable to reduce the number of features using feature selection techniques or dimensionality reduction. Features can be selected using techniques such as ANOVA and searchlight, and the techniques available for dimensionality reduction are singular-value decomposition and principal component analysis. Searchlight is a technique developed by

[1] Figure 1.1 and Figure 1.2 are reproduced from [55].

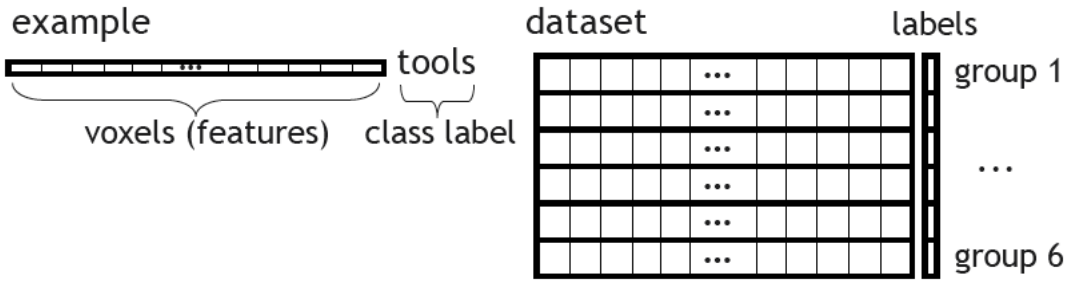


Figure 1.1: The left part of the figure shows voxels as features, and the right part of the figure shows a data set, that consists of group of such features stacked on top of each other.

Kriegeskorte et al. [37] in which a sphere of user-specified radius, having a center at one voxel, is moved throughout the brain and a multivariate statistic is computed at each location, with the result stored per voxel. This technique can be used to select different voxels.

Once the data set is divided into training and testing sets, a technique called cross-validation can be applied to compute the accuracy in order to assess how the classifier is performing. Usually leave-one-out cross-validation or k-fold cross-validation is performed. As the name suggests, in leave-one-out cross-validation, one of the examples is left out, and the classifier is trained on the other samples. This step is repeated for all the examples and the accuracy is computed across all the folds. k-fold cross-validation is similar to leave-one-out validation where the data set is divided into k subsets, and during each fold one subset is chosen as the testing data, and the classifier is trained on the remaining $k - 1$ subsets. Linear classifiers such as linear support vector machines and Gaussian Naive Bayes (GNB) are mainly used for classification of fMRI data.

Our study is divided into three related problems. The first two problems focus on analyses of fMRI data using different machine-learning and statistical methods, and the last problem focuses on developing a programming suite for easy and efficient analysis of the neuroimaging data

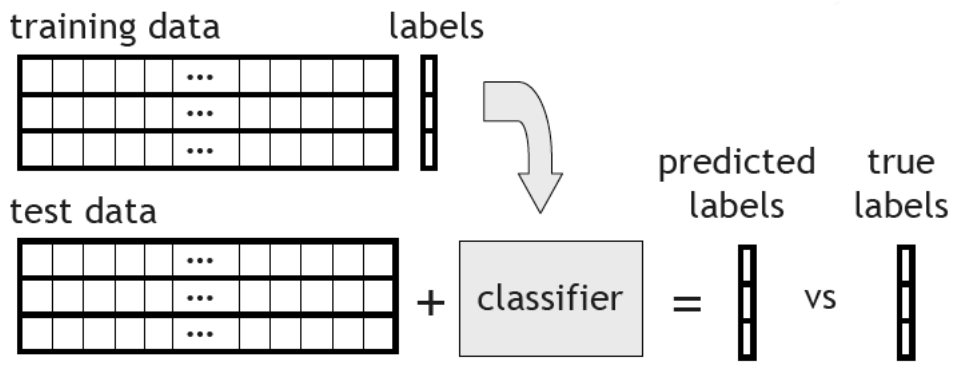


Figure 1.2: The figure shows how a machine-learning classification takes place. The data set is divided into training and test sets, and a classifier is trained on the training data set and tested with the test set.

The three problems are as follows:

1. Perform representational similarity analyses (RSA) of the fMRI data based on activation patterns from human brains. We project the results using multidimensional scaling (MDS) in order to visualize the similarity structure of the neuroimaging data. We also found out the best representational structure for the data set using an algorithm developed by Kemp et al. [34].

2. Analyze a sample fMRI data set with a non-hierarchical machine-learning classifier such as linear support vector machines (SVM), and a hierarchical machine-learning classifier algorithm such as a decision tree [68].

3. Develop a programming environment for the PyMVPA (Python Multivariate Pattern Analysis) [22] toolbox. PyMVPA provides many numerical algorithms for multivariate fMRI analyses. Our goal is to develop a wrapper tool around the PyMVPA toolbox so that analyses can be carried out by typing a small set of simple commands. The users should not have to learn Python programming to use our tool.

   We illustrate our analysis with an fMRI data set obtained using auditory stimuli.

We have designed a preliminary version of the wrapper tool equipped with all the necessary utilities to perform fMRI data analyses. We applied a multivariate searchlight technique [37] with a linear SVM classifier on the data set. The regions that code for our auditory inputs are mostly found in the temporal regions of the brain. We analyzed the data set with the decision-tree algorithm, and we provide a comparative analysis with the results obtained using SVM.

We have also computed and analyzed the dissimilarity matrices for the entire brain and different ROIs (Region of Interest) using the RSA method. The ROIs are the different areas within the brain such as inferior occipital area, middle temporal area, lingual area, frontal lobes, and so on. We have computed various representational structures for the data set using the Kemp's algorithm. The different basic representational structures are ring, tree, hierarchy, order, chain, and partition. These structures are generated using graph grammars and are used to represent different data sets such as similarity data, feature data, and relational data. The structures consists of nodes and links, and the each of the nodes represent a class or a label. The links indicate how the different nodes are clustered or grouped together. We found that our data set is best represented as a "ring" or a "tree" in different ROIs of the brain (For more information about Kemp's algorithm, please refer to Section 2.2.) "Tree" structure is voted the second best structure in the cases where the representation is a "Ring" structure, and "Ring" is voted the second best representation where "Tree" is the first preferred structural form.

The document is organized as follows. Chapter 2 gives background of the various techniques used in this study, Chapter 3 explains the data set, Chapter 4 provides the approach and the results, and Chapter 5 provides the related work and discussion.

Chapter 2

Background

In this chapter, we give an overview of the methods used in our study. The interested reader can refer to the relevant references to find more information about the different topics.

2.1 PyMVPA

PyMVPA [22] is a Python module used for performing multivariate pattern analysis (MVPA) [52] of large data sets. It is entirely written in Python and is an open-source software. Figure 2.1 shows the PyMVPA framework,[1] and how PyMVPA unifies various machine-learning algorithms into one package. PyMVPA is portable i.e., it runs on any platform that supports the Python programming language.

PyMVPA takes data, mask, and attribute file as inputs. The most common data and mask file format for PyMVPA is NifTI (Neuroimaging Informatics Technology Initiative) [20] format, and the attribute file can be in text format. The attribute file has the information about different labels and scans; fMRI data set is usually in four-dimensional form, and PyMVPA provides a mapping function to convert the data set

[1] Figure 2.1 is reproduced from [21].

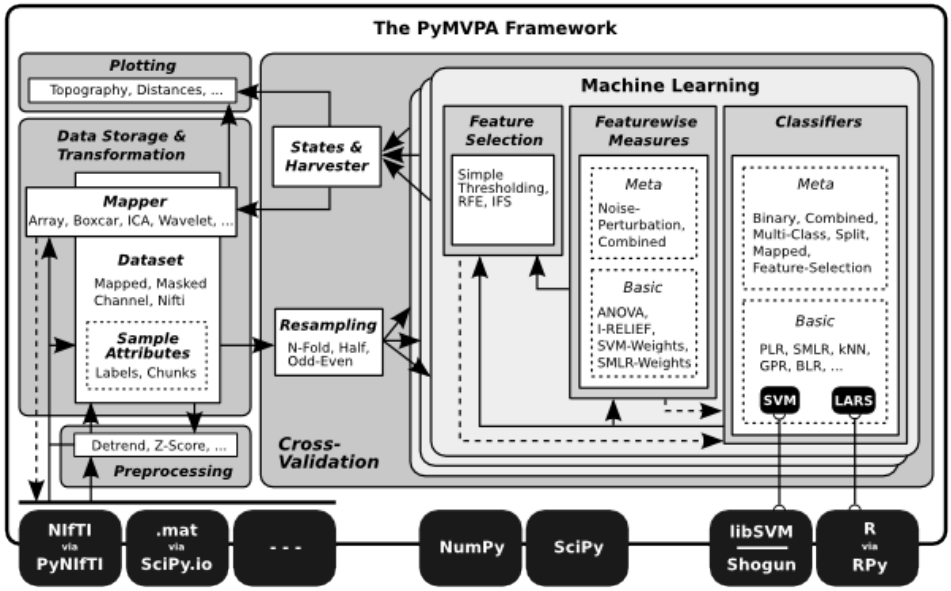


Figure 2.1: PyMVPA framework indicating the unification of different machine-learning algorithms, basic feature-wise measures, and meta-algorithms such as multi-class classification. The input file format for the data set can be NifTI or MATLAB (.mat) file format. PyMVPA provides cross-validation techniques and preprocessing functionality. PyMVPA data sets can be easily processed with SciPy and NumPy software packages.

into two-dimensional matrix consisting of $samples \times features$ for easy processing. It provides a reverse mapping function to convert the data set back into the original four-dimensional space. PyMVPA has many classifiers such as k-nearest-neighbor, linear SVM, sparse multinomial logistic regression, and other variations of regressions. It provides different feature selection methods, and also the multivariate searchlight method. We use this toolbox for a few of our analyses.

2.2 Representational similarity analysis

Systems neuroscience consists of finding a relationship between brain-activity measurement, behavioral measurement, and computational modeling. Since this procedure is highly complex, Kriegeskorte et al. came up with a new technique called representational similarity analysis in that gives information about the representation of the data

in brain [38].

We want to measure how dissimilar the categories are based on activation patterns of the human brain. RSA helps us to compute representational dissimilarity matrix (RDM). RDMs have a number that denotes how dissimilar the activity patterns associated with the two conditions are. An RDM is a symmetric matrix about a diagonal of zeros, and the measure of dissimilarity is typically a correlation distance. RDMs show which distinction between stimuli are emphasized and which are not. Unlike pattern-information analysis (Section 2.3), RSA does not involve classification of the data set, but it tests the activity patterns together.

Figure 2.2 shows the computation of a sample dissimilarity matrix.[2] The experimental conditions such as faces and houses are the input stimuli presented to a subject or a model. The activity patterns are the responses obtained to these input stimuli from brains or different models of the brain. These activity patterns are then used to compute the dissimilarity matrix. The matrix in this figure shows correlation between two faces and two houses. Blue indicates strong correlation between the data and red indicates no correlation. The dissimilarity matrix contains $1 - correlation$ values. Hence, it has all the diagonal elements as zeros. This matrix is symmetric about the diagonal. In the figure, we observe that faces are more correlated with each other, and houses are more correlated with each other. But we see much less correlation between the faces and houses. This result indicates that similar input conditions are highly correlated within the human brain. In RSA, the underlying represented information is compared and not the activity patterns themselves.

We perform an RSA [38] on the data set to measure the dissimilarity among the different auditory categories.

RSA consists of the following steps:

[2]Figure 2.2 is reproduced from the similarity analysis paper by Kriegeskorte et al. [38].

1. Measure the blood-oxygenation-level-dependent (BOLD) activity patterns using General Linear Model (GLM), or just have the raw fMRI data.

2. Measure the representational dissimilarity matrix (RDM) by computing the correlation between different categories of fMRI data.

3. Compute the dissimilarity matrix of fMRI data with a range of models of the human brain.

4. Compare the two dissimilarity matrices obtained in steps 2 and 3.

5. Test the two matrices by randomizing the labels.

6. Visualize the RDMs using MDS.

Finding structure in psychological data

The model developed by Kemp et al. [34] learns the structures of many different forms and finds out the best form for a given data set. This model discovers the underlying structure of the data using probabilistic techniques over a space of graph grammars and uncovers the structures of various forms such as trees, hierarchies, rings, and similar other forms. As we mentioned earlier, these structural forms consists of nodes and edges. Each of the nodes indicate a label. The edges indicate how these different labels are clustered together. If the data is a similarity data, the clustering indicates how similar the labels are when compared with the other labels. These structures are generated using graph grammars in which a context-free production is applied to a parent node to produce two child nodes. Each of these structures such as tree and ring have a specific set of rules as to how to connect the parent node to the child node and to the neighboring nodes (For more information please refer to [34].)

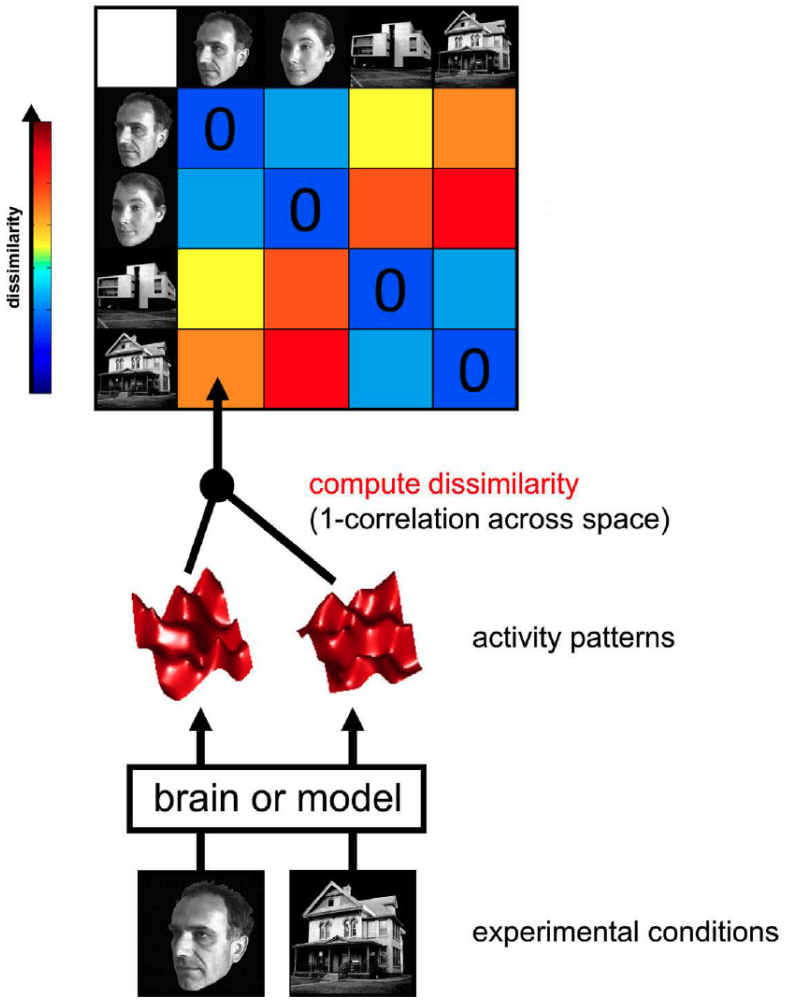


Figure 2.2: Representational dissimilarity matrix (RDM) of four experimental conditions. The dissimilarity between each pair of the experimental conditions is obtained by computing 1 minus the correlation of the activity patterns. These values are assembled in an RDM, a symmetric matrix about a diagonal of zeros.

Figure 2.3 shows the tree and ring[3] structure generated by the algorithm. The data used to generate the tree structure are features of different animal species, and the data used to generate the ring structure are similarity values obtained using human judgement on different colors of the color spectrum. Kemp's algorithm accepts a data matrix containing features or a dissimilarity matrix as an input.

Figure 2.3(a) shows how similar animals are clustered together. The herbivores such as elephant and giraffe are grouped together, the carnivores such as lion and dog form another cluster, insects such as ant and cockroach form the third group, birds such as robin and chicken form another cluster, and sea animals such as dolphin and penguin are placed near each other. According to the authors, when more features are added, we see improvement in the representational structure. Initially, they start with a few features of different animals and that yields "partition" as the structure. As more features are added, they obtain the tree structure shown in this figure. The authors say that the evolution in a human takes place in this manner. As a human learns new features, the representation of a particular data becomes more complex, but at the same time it yields a better representational structure for the data. Furthermore, the authors claim that given the data set for animal kingdom, intuitively one can think of representing it as a "tree", and their algorithm uncovers this structure.

Figure 2.3(b) shows the clustering of similar colors together. The red colors form a group, the green hues form another group, and all the blue hues are clustered in a group. The authors use the similarity measures obtained using human judgements on 14 pure-wavelength hues as an input to their algorithm. They say that the structure obtained from their algorithm corresponds to the Newton color circle.

We would like to apply this algorithm to an fMRI data set to see what patterns are likely to be uncovered by this algorithm. As far as we know, we are the first to perform this analysis on fMRI data. We will compute the dissimilarity matrices for our auditory

[3]Figure 2.3(b) is reproduced from Kemp et al. [34] paper.

data set and apply this algorithm to these matrices.

A brief overview of Kemp's algorithm is as follows: Given the data D, the model strives to find the form F and the structure S of that form that maximizes the posterior probability given by the following equation:

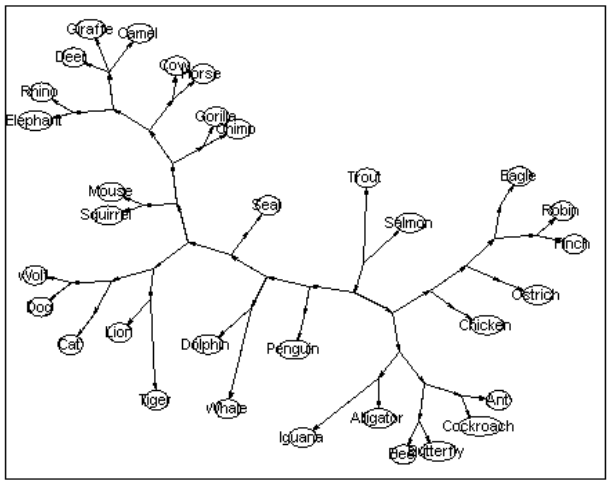
$$P(S, F|D) \propto P(D|S)P(S|F)P(F).$$

$P(F)$ is a uniform distribution over the various forms. $P(S|F)$ is the prior that favors a graph with least number of clusters. $P(D|S)$ measures how well the structure S accounts for the data D. The model accepts the dissimilarity matrices as the input, and hence we can process the dissimilarity matrix obtained using the RSA method using this technique. The algorithm works as follows. All the entities are assigned to a single cluster initially. Graph grammars are used to split these clusters into multiple clusters. The assignment of the entities to the split clusters is done using a greedy approach to maximize the posterior probability.

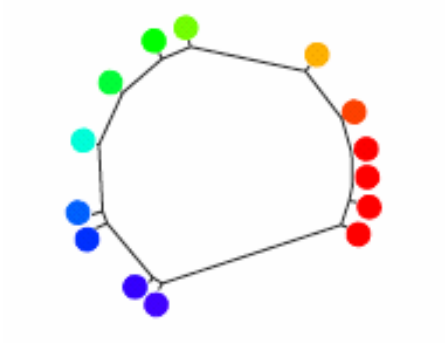
2.3 Pattern information analysis

Pattern information analysis [51] is a method to study how voxels code for a particular stimulus. Unlike activation based analysis, where only the regional activation of voxels is taken into account, pattern information analysis focuses on the representational content of the data. Activation based analysis involves spatial smoothing of data and averaging of activity across the voxels to find the brain regions that become activated as a whole for a particular stimulus. This averaging technique is called regional-average activation. But this technique results in loss of fine-grained spatial information. Pattern information analysis helps to overcome this problem.

A brief overview of this method is as follows:



(a) Tree



(b) Ring

Figure 2.3: Figure 2.3(a) and Figure 2.3(b) show an example of a tree and ring structure. The tree structure is generated using the feature values of different animal classes as the input. The ring structure is obtained using similarity matrices of 14 pure-wavelength hues as the input.

1. Select the training and test data and preprocess.

2. Estimate single-subject patterns—use GLM or raw data using single-volume intensity.

3. Select the voxels in training data—for example, using searchlight.

4. Train the classifier.

5. Test the classifier.

Since machine-learning classifiers enable us to carry out pattern localization, pattern discrimination, and pattern characterization, we will use pattern information analysis in this paper. The motivation for using this method in this research comes from the findings of Raizada et al. [56] that pattern information analysis can detect patterns even when there is no change in regional-average activation. In this study, the authors show that speech phonemes such as "ra" and "la" produce the same amount of activation in the brain. Using activation based analysis we cannot distinguish between these two sounds, since we consider only the difference in amount of regional-average activation. Hence the authors use pattern information analysis, to uncover the underlying representational content of the underlying voxels, that can successfully distinguish these two sounds.

2.4 Decision tree

A decision tree [68] is a classifier that uses a decision at each node to decide whether it is a leaf node or requires further splitting (recursive top-down approach). The goal is to construct a shallow tree that minimizes the impurity at the leaf node. The different impurity measures are entropy, misclassification error, and Gini. Whenever a split has to be made, the split that yields the least impurity is chosen. Splitting stops when the impurity value reduces below a certain threshold or only a few examples are left to split.

CART

CART (Classification and Regression Trees) is a decision-tree algorithm developed by Breiman et al. [7]. A CART is a binary tree constructed by repeated splitting of nodes starting from the root node that has the whole sample. The goal of CART algorithm is to choose a split that yields a child node with lowest impurity. CART uses Gini impurity measure [76], [69], and [24] to perform splitting at each node.

In this project, we analyze our data set with CART algorithm.

Chapter 3

Data set

The fMRI data used in the current experiments was obtained using different auditory stimuli.[1] In the following sections we explain the data acquisition and preprocessing steps in detail.

3.1 Stimuli

The stimuli consist of 40 different sounds, 20 of which belong to the animate category and the remaining 20 belong to the inanimate category. The stimuli are obtained from a commercial sound-effects library, Hollywood Edge. The animate category consists of four classes of data—bird chirps, dog barks, horse whinnies, and human voices.[2] The inanimate category also consists of four classes of data—car engines, gun shots, helicopter rotors, and phone rings. There are five exemplars per class of data; for example, the bird class consists of bird1, bird2, bird3, bird4, and bird5, as shown in Figure 3.1. Animate and inanimate categories together are referred to as the superordinate category. The different subclasses within the superordinate category are referred as basic

[1] This data set was obtained by Yune-Sang Lee.

[2] Human voices consisted of a crowd of human voices and was not language driven i.e., the sounds did not consist of one person talking but a group of people talking at the same time, and we cannot identify the language these people are talking in. These sounds were chosen as stimuli so that the subjects' responses are not biased by accent or language.

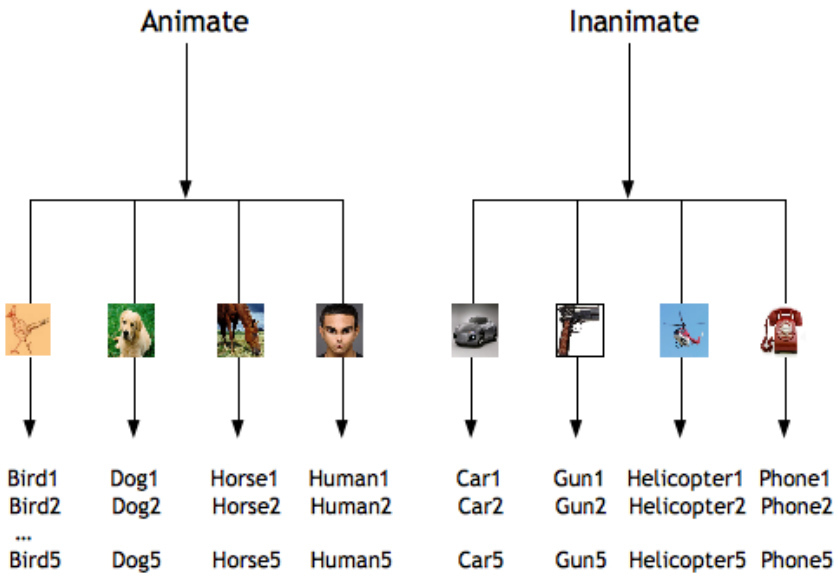


Figure 3.1: The different hierarchies in the sound categories. Animate and Inanimate are the superordinate categories. Bird, Dog, Horse, and Human are the basic categories within the animate sound category. Car, Gun, Helicopter, and Phone are the basic categories within the inanimate sound category. Five exemplars are presented per basic category throughout the scan.

categories, i.e., the basic animate category consists of bird, dog, horse, and human, and the basic inanimate category consists of car, gun, helicopter, and phone.

3.2 Subjects

There are eight subjects in all, of which four are males and four are females. The subjects are in the age group of 19–42 years and did not have any hearing disorders.

3.3 Data acquisition

The data is collected from fMRI scans as follows. Figure 3.2 gives an outline of the data acquisition for one subject. The following steps are repeated for all eight subjects. In each scan, every subject had five sessions. During each session, the sounds are presented from one of the previously mentioned classes of sounds (e.g., bird, dog). There are eleven stimuli per session. Each stimulus is presented for 8 seconds (TR = 2 seconds) and along with the stimulus, the stimulus number was shown on the screen. Visual cues were provided in order to help the subjects to distinguish the different sounds and also identify the target sound. In order to find out whether the subjects were paying attention to the stimulus presented, on the eleventh stimulus, subjects are asked to indicate whether the current stimulus (target sound) has been presented previously by clicking a button. For half of the sessions, the eleventh stimulus was chosen from the previously presented ones, and for the remaining half, it was chosen from a completely new class, such as wind chimes or duck sounds. The subjects had an 8-second resting period between the sessions. In all, there are eight such scans (also known as runs/chunks).

Blocks	Sound	Run1	Run2 ...	Run8	Time(sec)
Block1	sound1	horse5	dog2	bird5	2
	sound2 ...	helicopter1	bird3	gun4	10
Block2	sound1	car5	phone4	dog2	98
	sound2	car3	gun5	bird4	106
...					
Block5	...				
	sound9	gun1	bird2	human5	450
	sound10	dog3	human5	car4	458
	target	gun1	phone3	dog5	466

Figure 3.2: A sample of the fMRI scanning technique involved in data collection. The figure shows the procedure of data collection for one subject. The first column indicates the different sessions (only three out of the five sessions are shown). The second column indicates the visual cues presented to the subjects in order to help them distinguish the different inputs. The visual cues enable the subjects to identify the target sound. Columns 3–5 indicate the stimuli for the different runs (only three out of the eight runs are shown). The final column indicates the onset time of each stimulus.

3.4 Data preprocessing

The data is preprocessed to account for motion correction, and is also spatially normalized [65]. The data is not spatially smoothed, which, if done, would result in elimination of fine-grained resolution of responses. Statistical Parametric Mapping (SPM) [49] software is used to preprocess the data. After preprocessing, the data is in NifTI (.nii) format. We have 1920 NifTI files per subject, corresponding to all the scans and sessions. These NifTI files are input to the PyMVPA toolbox for further analysis.

We carried out the following steps using PyMVPA for each subject. We de-trended and z-scored the fMRI data per chunk. De-trending the data helps to remove the influence of time, such as slow drifts from the fMRI scanner. We z-scored the data to normalize the intensities across runs.

3.5 Data samples and labels

PyMVPA required specific inputs, which are generated as below. The attribute file has two columns: the first column indicates labels, and the second column indicates runs. In the attribute file, we have 1920 rows, that corresponds to the total number of NifTI files. Here we explain how we label the data in the attribute file. We select the data from fourth, sixth, and eighth scans[3] because the hemodynamic response for the stimuli is maximum during these scans. For superordinate classification, the labels are 1 (animate) and 2 (inanimate). For basic classification within the animate category, the labels are 1 (bird), 2 (dog), 3 (horse), and 4 (human). For basic classification within the inanimate category, the labels are 1 (car), 2 (gun), 3 (helicopter), and 4 (phone). For the similarity analysis the labels are 1 (bird), 2 (dog), 3 (horse), 4 (human), 5 (car), 6(gun), 7(helicopter), and 8 (phone). We label the data in fourth, sixth, and eighth scans with one of the previously mentioned labels (depending on the classification hierarchy or similarity analysis), and we label the data from the second scan as 0. We also label the resting periods as 0. The 0 labels will be removed during our data analysis.

For the similarity analysis task, we averaged the data samples per label within one subject. For example, we averaged all the bird samples such as bird1, bird2, ...to a single class called "bird". We repeated this procedure for all the classes. Hence we obtained eight samples, each one being labeled as one of the basic classes such as bird, dog, ... We computed the correlation values across the eight different classes, and stored $1 - correlation$ values in a RDM. We repeat this procedure for all the eight subjects in order to compute eight different RDMs. After computing the dissimilarity matrices for all the eight subjects, we averaged all these matrices across all the eight subjects to get an 8×8 RDM.

We did not average the data set for classification task, however. We had 1200 sam-

[3]Each stimulus lasts for eight seconds. The stimulus is repeated every two seconds. We refer to this two second window as a scan.

ples for superordinate classification, and 600 samples for basic-animate classification, and 600 samples for basic-inanimate classification.

Chapter 4

Methods and Results

In this section, we explain the different methods applied to the data set, and provide an overview of our programming environment.

4.1 Discovery of representational structure

We computed RDMs for the different categories of data, in order to find similarity within the same superordinate category. This problem is motivated by the works of Kriegeskorte et al. [39] where the authors compute the dissimilarity between different categories of visual input in the inferior temporal cortex.

As we mentioned previously, in order to compute the dissimilarity matrix, we averaged each of the labels' data. This task was done per subject. We used the averaged data set to compute the dissimilarity matrix using the Spearman correlation coefficient r. The dissimilarity matrix is a 8×8 matrix for the eight categories: bird, dog, horse, human, car, gun, helicopter, and phone. The dissimilarity matrices obtained from all the eight subjects were averaged to generate a dissimilarity matrix that we used for MDS and structural form analysis.

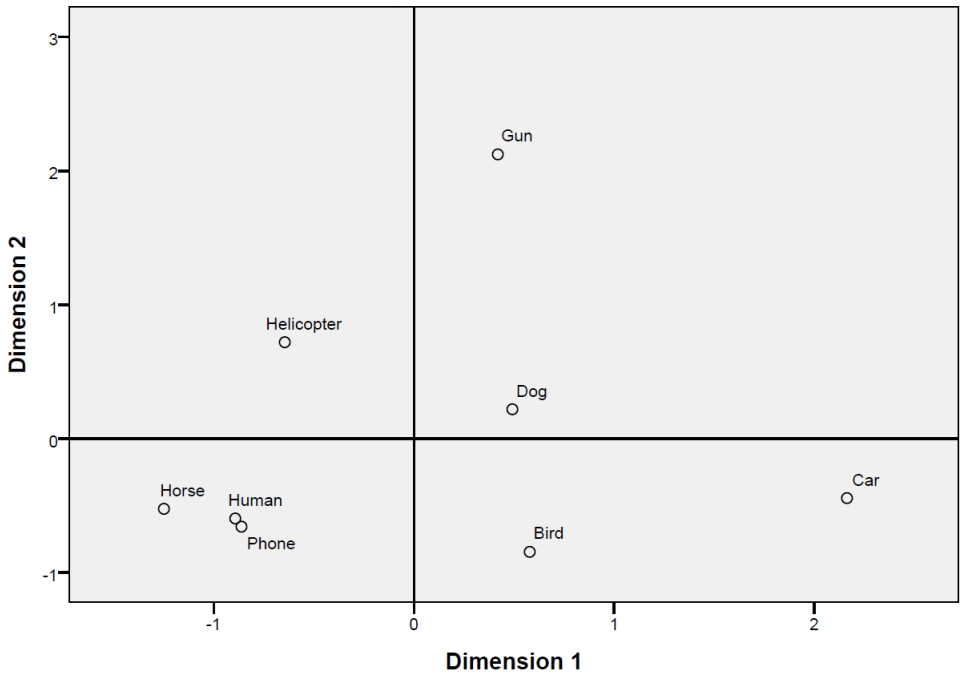


Figure 4.1: A two dimensional plot of MDS structure for the entire brain data. RDMs obtained using the entire brain data of eight subjects are averaged, and we applied a 2-D MDS algorithm on this data. The MDS values are computed along the first two MDS dimensions. We observe no clear separation between basic-animate and basic-inanimate categories.

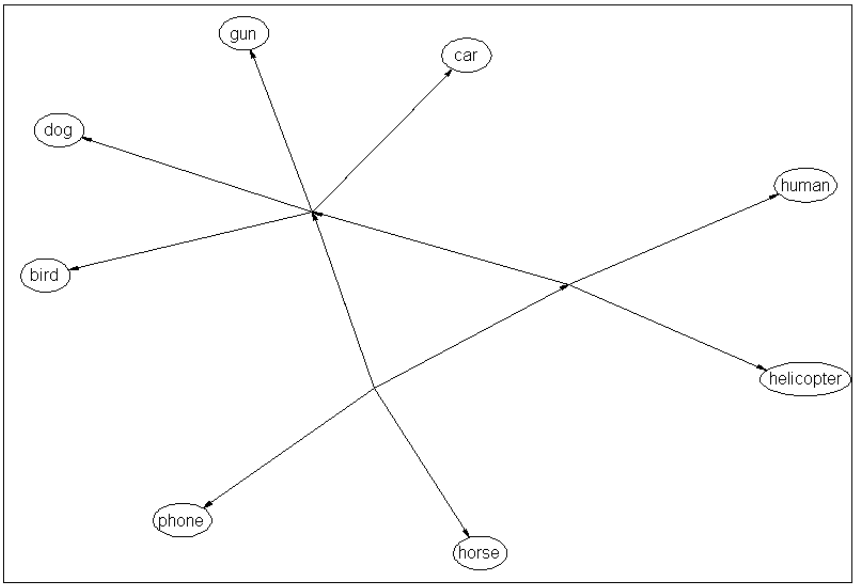


Figure 4.2: Ring structure obtained using Kemp's algorithm for the entire brain data. We used the RDM of the entire brain data averaged across all eight subjects as the input to Kemp's algorithm. The nodes represent the eight basic categories such as bird, dog, ... We do not observe significant clusters of basic-animate and basic-inanimate categories.

Figures 4.1 and 4.2 indicate our RSA results for the entire brain data. We observe no significant pattern or correlation among the different classes of the entire brain data. This result might be occurring due to the fact that the data obtained from entire brain is noisy. The noisy data is due to the presence of a large number of uninformative voxels in the data i.e., these voxels have a low signal-to-noise ratio. We can overcome this problem by selecting the voxels from different ROIs to check the if the particular ROI has interesting information. We performed RSA on different ROIs as well, and we obtained relatively better results when compared with the entire brain data. These results are presented later in this section.

Figure 4.1 shows a two dimensional plot of MDS structure for the entire brain data. RDMs obtained using the entire brain data of eight subjects are averaged, and we applied a 2-D MDS algorithm on this data. The MDS values are computed along the first two MDS dimensions. The correlation values in the dissimilarity matrix are subjected to a minimization problem that computes the stress values. The computed stress values are the difference between the points (x, y) in the N dimensional (2-D in our case) space and the original dissimilarity matrix values. The coordinates that yield the least stress values are chosen to be values for the particular class (For more details about MDS computation, please refer to [64]). In our result for the entire brain data, the basic-animate categories such as human and horse, and dog and bird are near each other. But all the basic-inanimate categories are spread out in the MDS space. In addition, we observe that the phone sound is very close to the human sound. These results could be due to the fact that the data obtained from the entire brain is noisy.

We computed harmonic-to-noise ratio (HNR) values of different auditory sounds. The HNR values denote the periodicity of the particular audio signal [50], [77]. High HNR values denote high periodicity, while lower values denote low periodicity. We observed that phone sound has HNR values similar to human sounds. In the figure,

we notice the phone sound is closer to the human sound. While this result could be purely due to noise, we also think this result could have occurred due to similar HNR values. The animate sounds have relatively high HNR values when compared with the inanimate categories. Phone sound is an exception, however. We computed the HNR values using Praat software [4].

Figure 4.2 shows a "ring" structure obtained using Kemp's algorithm for the entire brain data. We used the RDM of the entire brain data averaged across all eight subjects, as the input to Kemp's algorithm. The nodes represent the eight basic categories such as bird, dog, ... We do not observe significant clusters of basic-animate and basic-inanimate categories. Bird, dog, gun, and car form a cluster, human and helicopter form another cluster, and phone and horse form a cluster. These groupings do not convey good separability among various classes. "Tree" was voted to be the second best representation. Even in this form of representation, no striking patterns could be seen. We checked the other forms such as grid and hierarchy for separable patterns, but these structures exhibit more clutter than the tree and ring structure.

Since we did not observe good correlation patterns among different categories, we tried to improve the values by randomly leaving out 10% of the samples. The intuitive idea behind leaving out samples is to verify if the noise is occurring in all the samples, or just a few samples. We checked if the average score improved as a result of reducing the number of samples. We also left out different subjects while computing the average correlation matrix. The reason for leaving out subjects was again based on the hypothesis that one of the subjects' data could be noisy which in turn may affect our final averaged matrix. Unfortunately, neither of these approaches yielded any improvements. We also projected the RDM of individual subjects using MDS. These results were also noisy, and we did not observe clear separation among the different categories. Since we did not obtain good results despite reducing the samples, and leaving-out subjects,

our earlier hypothesis that the data obtained from the entire brain is noisy appears to be valid.

Hence, in order to test our hypothesis, we performed the same analyses on different ROIs of the brain to determine whether better results can be obtained. The ROIs[1] such as middle temporal regions and inferior occipital regions produce nice separation between the animate and inanimate categories, while few of the other regions such as insular and lingual regions do not produce any clear separation.

We think that the reason for better performance in the temporal regions as compared with other ROIs like lingual and insular regions is due to the higher response of voxels for auditory sounds in the temporal areas. On the other hand, results in the visual areas such as inferior occipital regions are interesting to observe because even though these areas are primarily concerned with vision, they display separability among the different auditory categories.[2] The entire brain data had a lot of noise to generate good results using similarity analysis. Hence, data obtained from various ROIs can be used for performing RSA to obtain satisfactory results.

Figures 4.3— 4.6 show the RSA results of the middle temporal and inferior occipital regions.

As we mentioned earlier, for computing the MDS representation, the RDMs (Please refer to the Appendix A for the dissimilarity matrices) obtained using the entire brain data of eight subjects are averaged, and we applied a 2-D MDS algorithm on this data. We compute the stress values between the original correlation values and the new co-ordinates, and choose those coordinates that yield the least stress values for the particular class (For more details about MDS computation, please refer to [64]).

[1]The ROIs we used for this analysis are obtained by Yune-Sang Lee using MarsBaR [8] software.

[2]Please refer to results in Section 4.2 for our hypothesis about visual areas coding for auditory sounds.

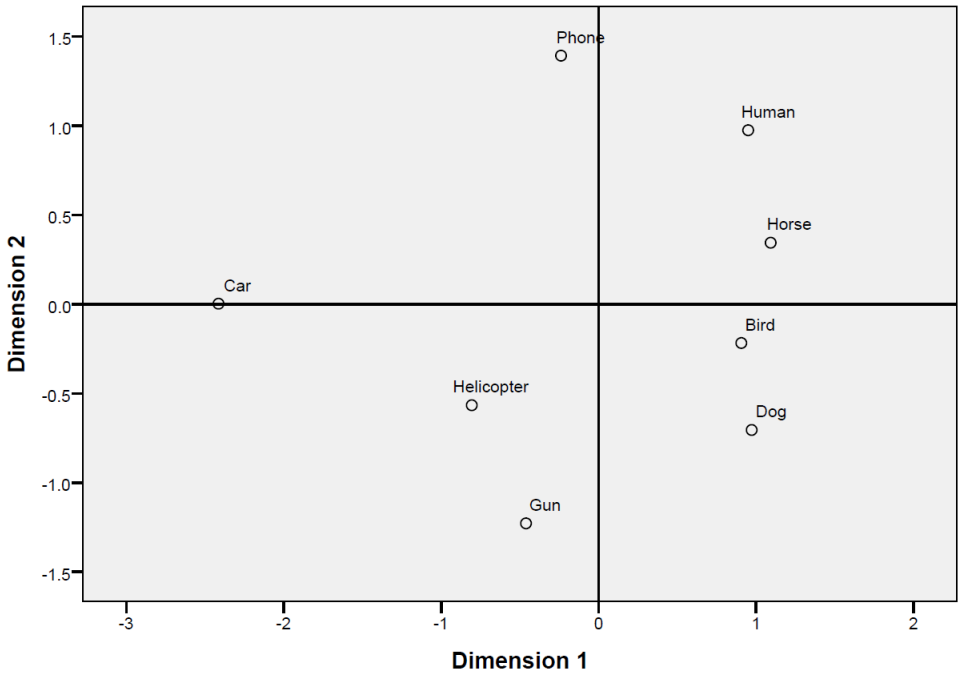


Figure 4.3: A two dimensional plot of MDS structure for the middle temporal area. RDMs obtained using the data from middle temporal area of the eight subjects are averaged, and we applied a 2-D MDS algorithm on this data. The MDS values are computed along the first two MDS dimensions. Unlike the entire brain data, in this MDS plot, we can see a separation between the animate and inanimate categories.

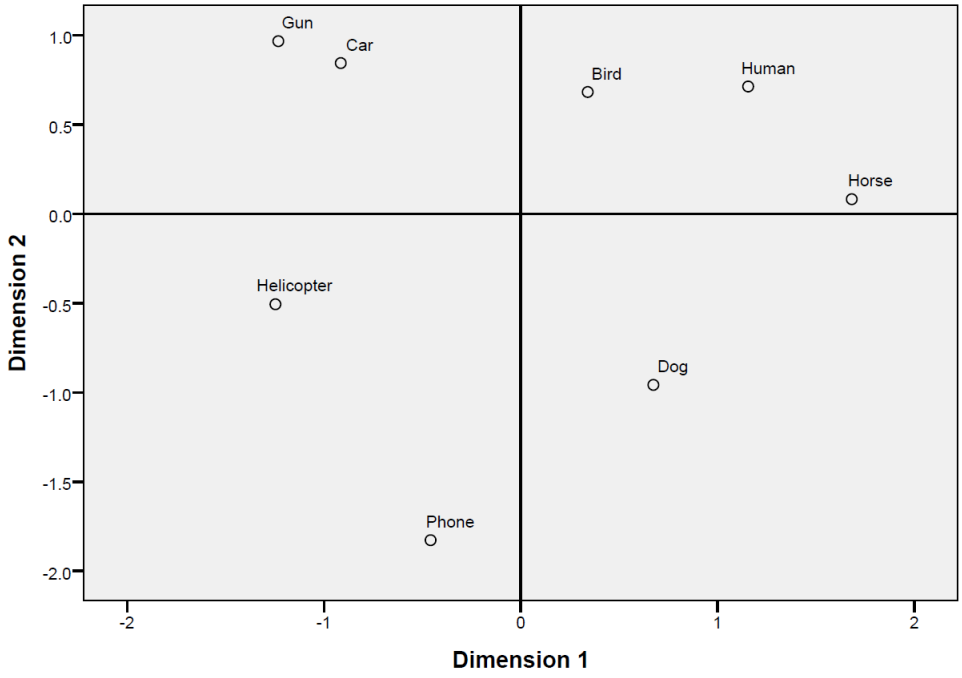


Figure 4.4: A two dimensional plot of MDS structure for the inferior occipital area. RDMs obtained using the data from inferior occipital area of the eight subjects are averaged, and we applied a 2-D MDS algorithm on this data. The MDS values are computed along the first two MDS dimensions. Unlike the entire brain data, in this MDS plot, we can see a separation between the animate and inanimate categories.

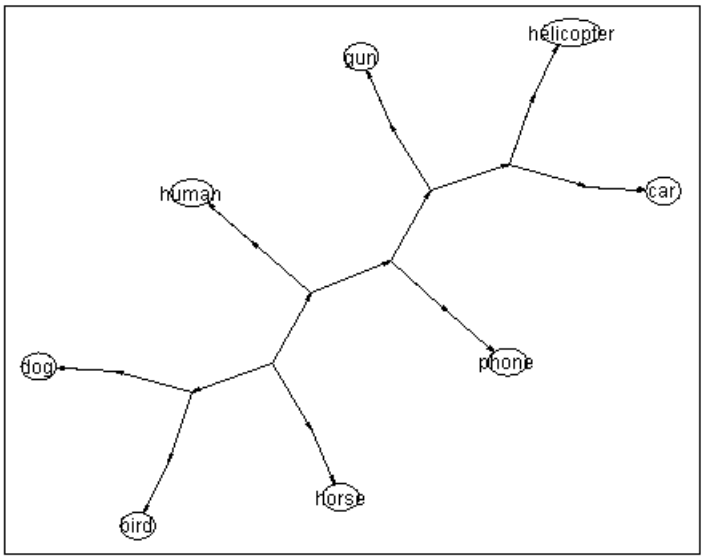


Figure 4.5: Structural form for the data obtained from middle temporal area. RDMs obtained using the data from middle temporal area of the eight subjects are averaged, and we applied Kemp's algorithm on this data. The nodes represent the eight basic categories such as bird, dog, ... We obtained a tree structure using Kemp's algorithm for the data from middle temporal area.

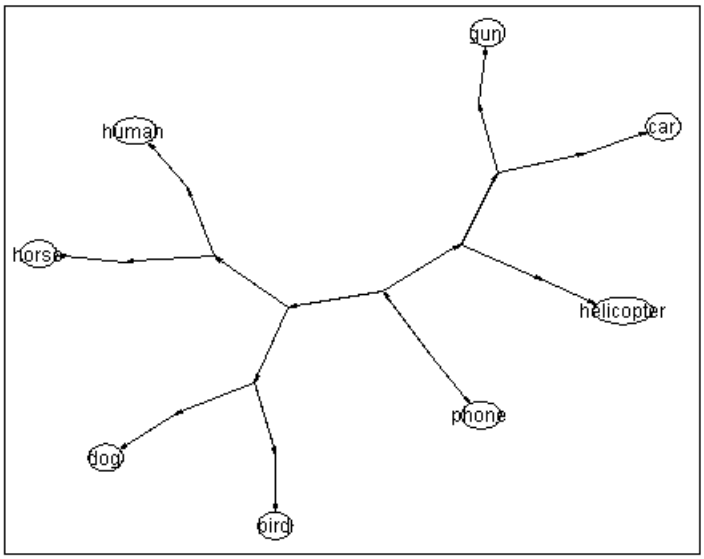


Figure 4.6: Structural form for the data obtained from inferior occipital area. RDMs obtained using the data from inferior occipital area of the eight subjects are averaged, and we applied Kemp's algorithm on this data. The nodes represent the eight basic categories such as bird, dog, ... We obtained a tree structure using Kemp's algorithm for the data inferior occipital area.

Figure 4.3 shows a two dimensional plot of MDS structure for the middle temporal area. Unlike the entire brain data, in this MDS plot, we can see a separation between the animate and inanimate categories. The basic-animate categories such as bird, dog, horse, and human are separated from the inanimate categories such as car, gun, helicopter, and phone along the second dimension. Within basic-animate group, we observe that bird and dog sounds are more similar to each other. All the animate sounds are placed near each other. Within the inanimate category, the sounds are more spread out. Car, gun, and helicopter sounds are more closer to each other. Phone sound lies more closer to the human sounds. We hypothesize HNR values of the phone sound and human sound could be the reason for this result. As we had mentioned previously, the phone and human sounds have similar HNR values. All the animate categories have high HNR values when compared with the inanimate categories.

The car sound is placed more to the left. This result is due to the fact that car is highly dissimilar from the animate categories while the other inanimate categories have correlation values relatively smaller than the car (see Figure A.2). Hence, car appears farther from all the animate categories. We would like to remind the readers that MDS is just a projection of the correlation matrix. The placement of the points are dependent on the dissimilarity values in the matrix. The algorithm generates the separating lines seen in the graph, and it is purely based on the correlation values in the dissimilarity matrix. The separation along the first dimension could be due to noise in the data. From the matrix, we can observe a few clear correlation patterns within the animate category. But the inanimate category do not seem to indicate clear correlation patterns in the matrix. Furthermore, we do not see the separation among the animate and inanimate sounds from the matrix. On the other hand, when we project our results using MDS, we observe how the different sounds are separated and what is the degree of similarity between the different classes of sounds.

Previous researches have shown that superior middle temporal regions are more

adapted to code for animate sounds than inanimate sounds [29]. These studies used GLM techniques for analysis. From our MDS results for middle temporal areas, we observe closer grouping of animate sounds, and also better correlation values within the animate category. Hence we hypothesize that perhaps middle temporal regions are indeed more suited for coding animate sounds, and we think that this coding depends on the HNR values.

Figure 4.4 shows a two dimensional plot of MDS structure for the inferior occipital area. Unlike the entire brain data, in this MDS plot, we can see a separation between the animate and inanimate categories. The basic-animate categories such as bird, dog, horse, and human are separated from the basic-inanimate categories such as car, gun, helicopter, and phone along the second dimension. The separation along the first dimension could be once again due to noise caused by averaging. We also notice the gun and car cluster together. Within the animate category, bird, human, and horse are grouped more closely together than the dog sound. If we inspect the dissimilarity matrix of this area (refer to Figure A.3), we observe that unlike RDM of the middle temporal area, inferior occipital area produces correlation patterns among both animate and inanimate categories. The placement of car and helicopter in the MDS representation seems to indicate they are more dissimilar even though they have good correlation values. But, we would like to remind the readers that, MDS aims to obtain good distances from all the points and not just one point. Hence the (x, y) co-ordinates that yield the least stress value from all the points are chosen.

Since this is a visual area, we hypothesize that the patterns we observe in this region are invoked by visual memories of the subjects, when they are responding to a particular sound (Refer to results in Section 4.2 for our hypothesis about visual areas coding for auditory sounds.) We think that the representation in MDS is not based on the HNR values of the sounds, but based on the representation of the visual content in the brain

for that stimuli.

We observe some similarity between the correlation values of the middle temporal and inferior occipital areas. Even though, the correlation values are not that great, we see a separation between the animate and inanimate categories along the second dimension for both the regions. Bird and dog seem to be more correlated for both the regions. We tried to improve the MDS representation by tweaking the stress value for convergence, but it did not yield significant improvements to our results.

We applied Kemp's algorithm on the RDMs obtained from the middle temporal areas and inferior occipital areas. These RDMs are obtained after averaging the RDMs across all eight subjects. We computed the posterior probabilities across all the forms and found "Tree" to be the form with the highest posterior probability. Initially, the algorithm starts off with all the classes in one cluster. In the subsequent steps, the graph grammars are applied repeatedly to generate the splits. The result of the algorithm is a form that has least number of clusters. In addition to the form, we also obtain posterior probability values using which we can choose the best form (Refer to Section 2 and [34] for more details about the algorithm).

Figure 4.5 shows the "tree" structural form for the data obtained from middle temporal area. The nodes represent the eight basic categories such as bird, dog, ... This representation of the data in this structure is similar to the MDS representation. We notice the separation among the animate and inanimate categories. Bird and dog sounds are form the sub-branch in the tree indicating they are more similar than the other animate sounds. Once again, phone is closer to human sounds. An interesting observation is that helicopter and car form a sub-branch in this representation, even though they are placed far apart in the MDS representation. This representation is due to the fact that, car is more dissimilar ($correlation > 1.1$) to all the animate categories. But, it is closer

to the helicopter in terms of the correlation values (Refer to Figure A.2). Hence we see this grouping, in this representation. We also looked at other structural forms obtained from this algorithm. "Ring" being the second best representation had the animate categories grouped on one side of the ring, and the inanimate categories were grouped on the other side of the ring. Bird and dog were placed closer to each other in this ring. Other forms of representation such as hierarchy, did not yield such clear separations.

Figure 4.6 shows the "tree" structural form for the data obtained from inferior occipital area. The nodes represent the eight basic categories such as bird, dog, ... In this structure, animate categories are grouped on one side of the tree, and the inanimate categories are placed on the other side of the tree. Within basic-animate category, we observe that human and horse form a cluster, dog and bird form another cluster. These groupings are due to the correlation values obtained from the matrix. Within basic-inanimate categories, we observe that car and gun are clustered together. This clustering of car and gun is similar to the MDS representation. While computing the structural form, the algorithm needs to account for the optimal placement of all the categories. Hence, we notice the car and gun are placed in a sub-branch even though car and helicopter sounds are more correlated.

We think that Kemp's algorithm yields a more intuitive representation of the RDM than MDS. When we think of a representation for our data set, we naturally choose a tree structure due to the hierarchical nature of our data set. We can also think of the data set to be a ring since we can group the classes around the ring. Another benefit of using the Kemp's algorithm is that it deals with the missing data effectively. We can tweak different parameters like prior values and the penalty value to improve the representation. For our data set, we did not gain much by altering these parameters, however. The results from Kemp's algorithm match with our hypothesis in our proposal. We predicted the most voted structural form for our data would be a "ring" or a "tree." From

these results, we conclude that any data set that has some form of grouping or hierarchy associated with it, can be best represented as a "tree" or a "ring."

Similar to entire brain RSA, we tried to improve the correlation values of the data obtained from middle temporal and inferior occipital area data by randomly leaving out 10% of the samples. We did not see any improvement in the results. We also left out different subjects while computing the average correlation matrix since we thought that a particular subject might be altering the average value. We left out different classes while computing the MDS and Kemp's representation to see if we obtained better results. But these approaches did not improve our existing results. The motivation behind using these approaches was to check if any particular data sample or subjects' data was noisy, and hence affecting our representation. We also performed analysis with individual subjects' data without averaging. While a few of the subjects yielded good representations in MDS, they cannot be considered a significant improvement to our results.

Rydell et al. [58] have proposed to average only the similar correlation values. We did not try this approach on our data set. This task could be an extension to this project to check whether we obtain better representation by selective averaging techniques.

We generated the dissimilarity matrices using PyMVPA and generated the MDS projections using statistics package of the SPSS software [25]. For more information about dissimilarity analysis, please refer to Appendix A. We generated the representational structures using Kemp's software [33].

4.2 Machine-learning classification algorithms applied to the auditory data set

We carried out analysis of the popular Haxby data set [28] with the two machine-learning classifiers, SVM and decision tree, before we applied them to the auditory data set. This analysis was carried out in order to verify whether the algorithms' produce similar result as Kuncheva et al's result [40] . We did obtain similar results as Kuncheva et al. where SVM performs better than decision-tree in terms of accuracy. We used ANOVA technique to select the top 1000 voxels before analyzing it with SVM. Decision trees do not require feature-selection because it selects only the best feature at each split, and it stops splitting based on the splitting criterion.

We applied linear SVM to our data set. We used the linear SVM implementation available in the PyMVPA toolbox. We carried out all the analysis on a 64-bit Linux machine equipped with a Intel(R) Core(TM) i7 processor and 12 GB RAM. The Linux version was Ubuntu 9.10.

As we previously mentioned in Section 3.5, for superordinate classification, the labels are 1 (animate) and 2 (inanimate). For basic classification within the animate category, the labels are 1 (bird), 2 (dog), 3 (horse), and 4 (human). For basic classification within the inanimate category, the labels are 1 (car), 2 (gun), 3 (helicopter), and 4 (phone). We used this labelled data set to perform our analysis. For superordinate classification, we had 1200 samples, and for each of the basic-animate and basic-inanimate classification we had 600 samples.

Figure 4.7 shows the flow diagram of the pattern classification method explained in Section 2.3. We divided the fMRI data into training and testing sets. Of the eight runs in the data, the training set consisted of six runs and the testing set consisted of two runs. We trained an SVM classifier with a linear kernel on the training data set. We used the

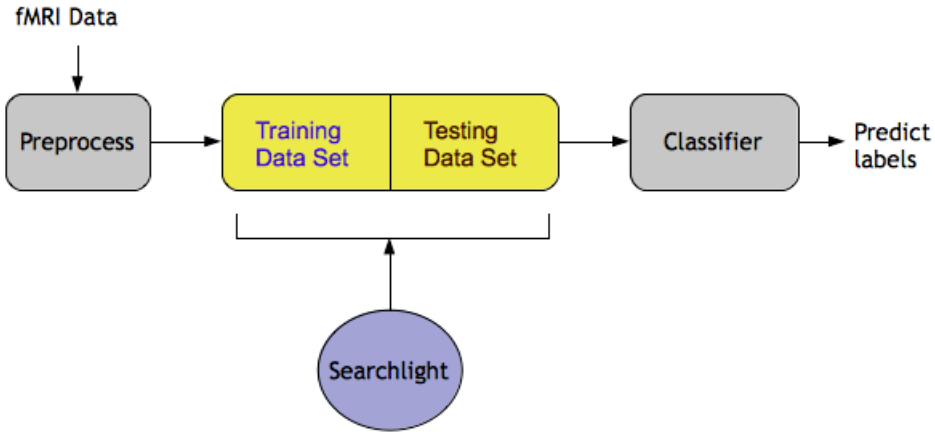
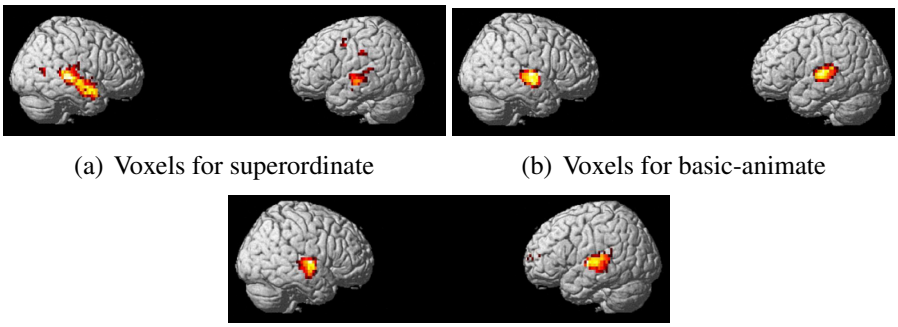


Figure 4.7: A flow diagram indicating the basic steps in the analyses. We divided the preprocessed data set into training and testing sets and trained a linear SVM classifier on the data set using searchlight. We tested our trained classifier on the testing data set using searchlight, calculated the percent error, and stored the error per voxel.

searchlight technique to select the voxels from the brain. A sphere of radius 11 mm was used to select about 120–140 features (voxels). We repeated this step four times, yielding a four-fold cross-validation. We averaged the results across the four-folds and subjected the accuracy values to a second-level random effect analysis [43], [54]. We performed the random effect analysis using MATLAB and SPM5 [49].

PyMVPA supports both two-class and multi-class classification for SVM. We could not perform finer levels of computation, such as classification within the basic categories, due to limited data.

Figure 4.8 show the results of applying searchlight and random effect analysis on the auditory data set. For basic-animate categories the responses are found in superior and middle temporal regions, and in Heschl's gyrus. For basic-inanimate categories we uncovered voxels in superior and middle temporal regions, and in calcarine regions. For superordinate category, the voxels appeared to be primarily in superior and middle temporal regions. In addition, we found voxels in the inferior occipital regions cod-



(a) Voxels for superordinate (b) Voxels for basic-animate

(c) Voxels for basic-inanimate

Figure 4.8: Figure 4.8(a) shows the voxels obtained for superordinate category. Figure 4.8(b) shows the voxels obtained for basic-animate category Figure 4.8(c) shows the voxels obtained for basic-inanimate category. We obtained 531 voxels for basic-animate category, 646 for basic-inanimate category, and 737 voxels for superordinate category. Superior temporal and middle temporal regions are the most activated areas for the auditory sounds. We can see a few voxels in the occipital and frontal lobes as well.

ing for the auditory stimuli. We also found small amount of voxels in frontal inferior triangularis regions.

We can observe overlap in some of voxels that code for basic animate and inanimate categories. This result is similar to the works of Staeren et al. [62] where the authors find overlapping responses in the auditory cortex. We also found some voxels that code exclusively for animate and inanimate categories. This result is similar to the works of Lewis et al. [29] where the author shows category specific regions were discovered. For superordinate category, some of the voxels overlap with the basic categories, but a few exclusive voxels were found to code for superordinate sounds in the temporal-occipital boundary.

We uncovered 531 voxels coding for basic-animate category, 646 for coding basic-inanimate category, and 737 voxels coding for superordinate category. These results are comparable to the results obtained by Yune-Sang Lee [44].

The voxels in the temporal areas are expected to code for auditory sounds. It is interesting to observe a few of the non-temporal areas code for auditory sounds as well.

We hypothesize that the areas known for responding to visual inputs such as the occipital regions are uncovered in this study due to "mental imagery". This phenomenon might be happening because when the subject is listening to an audio stimuli, there might be some visual memories being invoked, which in turn might result in activity in the visual area and frontal lobes [36]. It could also be the case that these areas respond selectively to certain visual or auditory inputs [42]. As yet, the correspondences between voxels representing images and sounds corresponding to a particular entity are poorly understood.

We analyzed the auditory data set with a decision-tree algorithm. We used the decision tree implementation available in MATLAB [47] to run our analysis. This implementation is a CART algorithm (Section 2.4). We performed a four-fold cross-validation on the data set using this algorithm.

We obtained similar accuracy values for our data set using both SVM and decision tree. This result is different from what is observed by Kuncheva et al. for Haxby data set, in which the authors claim that SVM produces better accuracy than decision tree [40]. We think that the difference between the accuracy values obtained from various classifiers depend entirely on the data set. The neuroscientists can try both the algorithms to decide what is suitable for their data set.

Figure 4.9—Figure 4.14 show the receiver operating characteristic (ROC) plots [67] for the three levels of classification. The ROC plots generated using SVM and decision tree are similar for both superordinate and basic levels of classification.

ROC plots are obtained using false positive rates (FPR) and true positive rates (TPR). TPR, also known as *sensitivity* measures the proportion of correctly classified true positive values. FPR is measured as $1 - specificity$ where *specificity* measures the proportion of correctly classified true negative values.

The x-axis of the ROC plot is FPR, and the y-axis of the plot is TPR. Perfect classi-

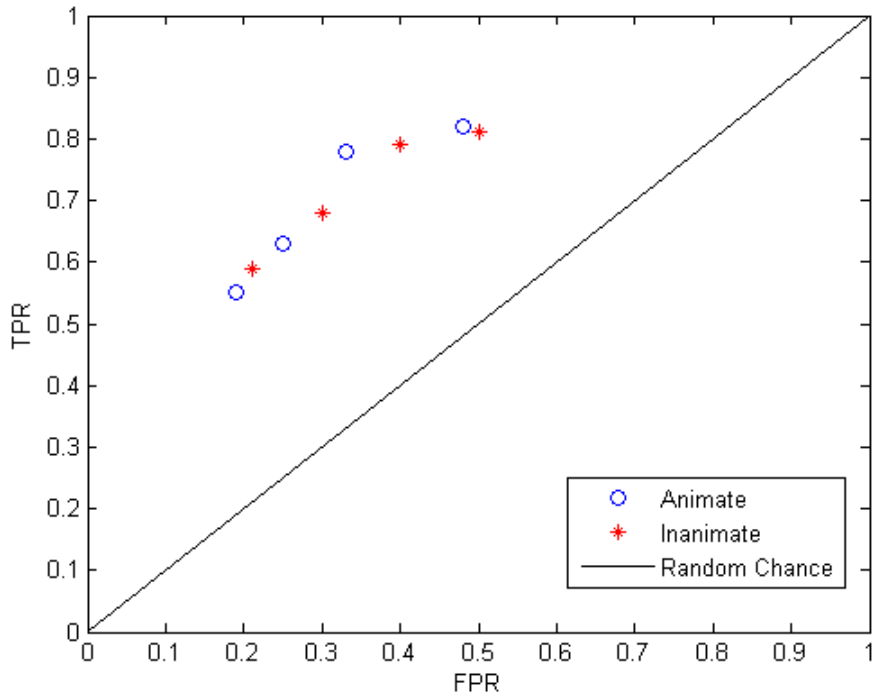


Figure 4.9: ROC plot for superordinate level of classification using SVM. The x-axis denotes the false positive rate (FPR) and y-axis denotes the true positive rate (TPR). We average the TPR and FPR for each superordinate category i.e., animate and inanimate across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular category. The different TPR and FPR values are obtained by varying the regularization parameter. The circle marker in blue color denotes the values for animate category, and the "*" marker in red color denotes the values for inanimate category. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line. We observe that all our values are above the random-chance line.

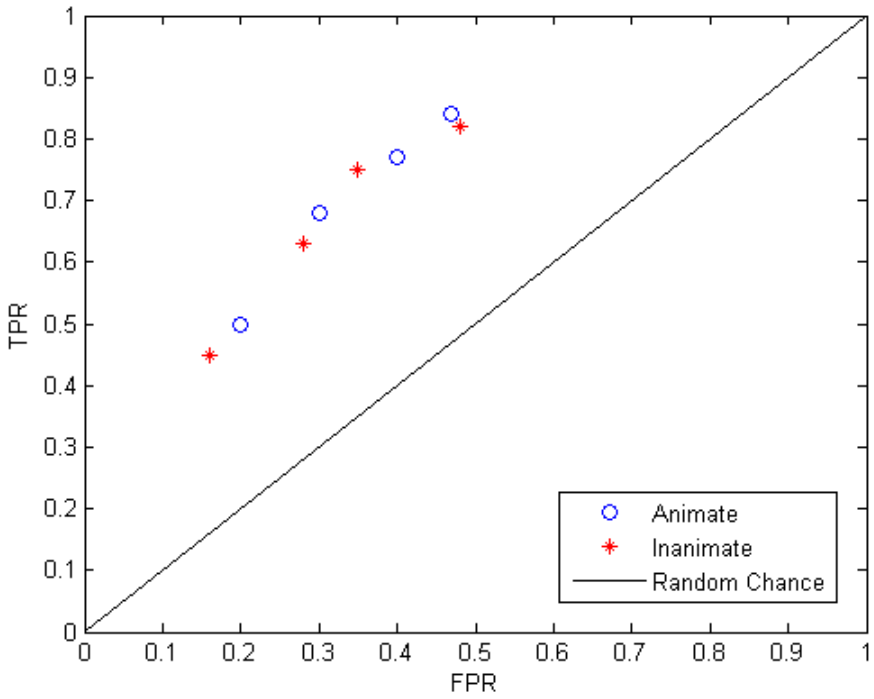


Figure 4.10: ROC plot for superordinate level of classification using decision tree. The x-axis denotes the false positive rate (FPR) and y-axis denotes the true positive rate (TPR). We average the TPR and FPR for each superordinate category i.e., animate and inanimate across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular category. The different TPR and FPR values are obtained at different levels of pruning. The blue circle marker denotes the values for animate category, and the red "*" marker denotes the values for inanimate category. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line. We observe all our values are above the random-chance line.

fication occurs when the points in the ROC plot lie at $(0, 1)$ i.e, the TPR values should be as high as possible (close to one), and the FPR values should be close to zero. Random guesses would yield points on the diagonal line connecting $(0, 0)$ and $(1, 1)$. Any point lying below the random-chance lie represents poor classification, and points above the random-chance line denote good classification. The TPR and FPR values can be computed from a confusion matrix.

From the ROC plots we observe that, all the values are above the random-chance line. This result indicates that these values have predictive accuracy better than false alarms. Any values on the chance line would indicate that the categories were not significantly detected. We do not observe such misclassification in our ROC plots. We provide more details about each of the graphs in the following passages.

Figures 4.9 and 4.10 shows the ROC plots for superordinate level of classification for SVM and decision tree respectively. We average the TPR and FPR for each superordinate category i.e., animate and inanimate across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular superordinate category. The blue circle marker denotes the values for animate category, and the red "*" marker denotes the values for inanimate category. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line.

Figures 4.11 and 4.12 shows the ROC plots for basic-animate level of classification for SVM and decision tree respectively. We average the TPR and FPR for each basic-animate category i.e., bird, dog, horse, and human across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular basic-animate category. The blue circle marker denotes the values for bird, the red "*" marker denotes the values for dog, the magenta square marker denotes the values for horse, and the black star (pentagram) marker denotes the values for human. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line.

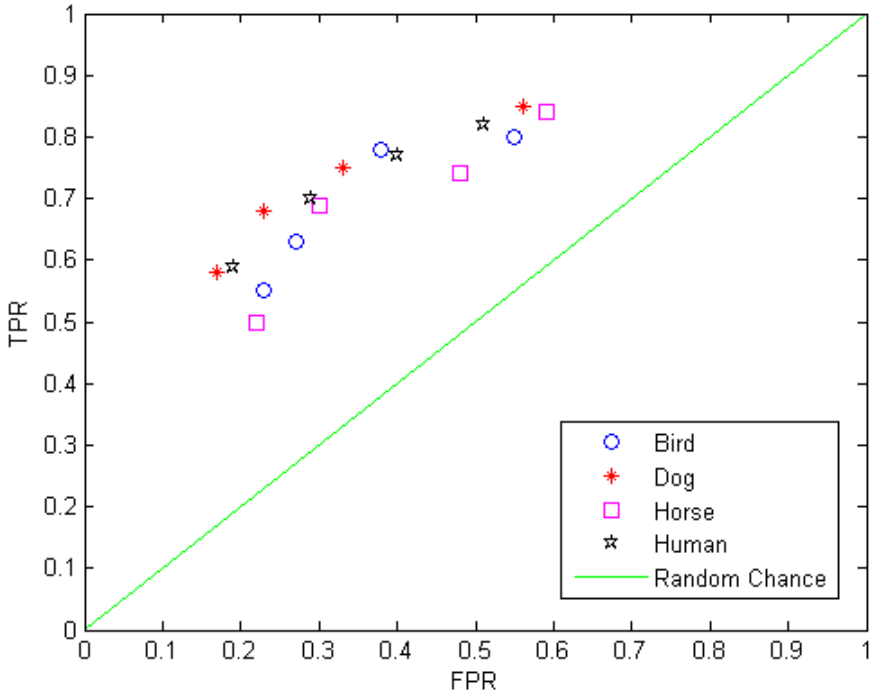


Figure 4.11: ROC plot for basic-animate level of classification using SVM. The x-axis denotes the false positive rate (FPR) and y-axis denotes the true positive rate (TPR). We average the TPR and FPR for each basic-animate category i.e., bird, dog, horse, and human across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular class. The different TPR and FPR values are obtained by varying the regularization parameter. The blue circle marker denotes the values for bird, the red "*" marker denotes the values for dog, the magenta square marker denotes the values for horse, and the black star (pentagram) marker denotes the values for human. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line. We observe that all our values are above the random-chance line.

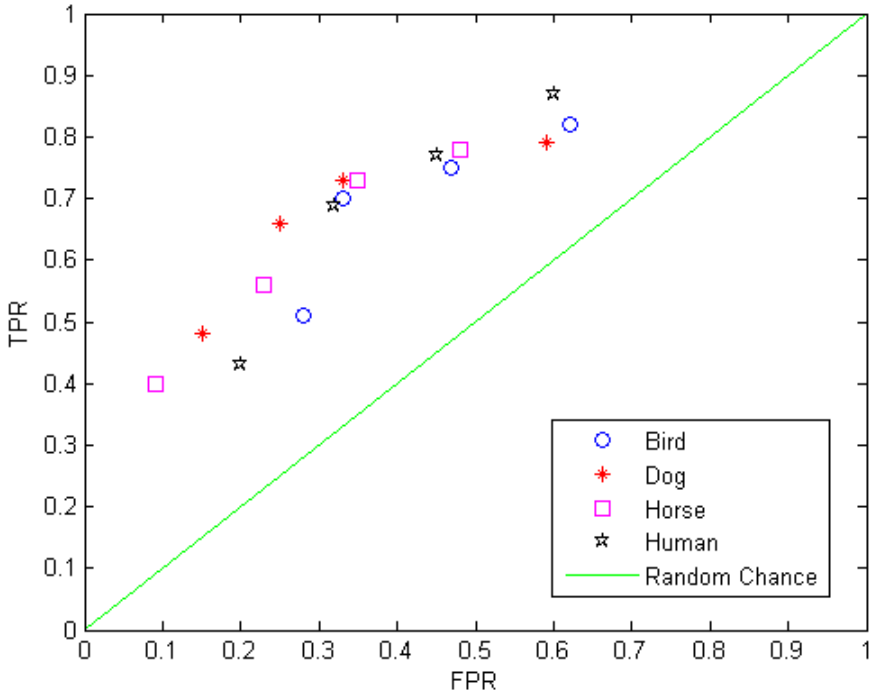


Figure 4.12: ROC plot for basic-animate level of classification using decision tree. The x-axis denotes the false positive rate (FPR) and y-axis denotes the true positive rate (TPR). We average the TPR and FPR for each basic-animate category i.e., bird, dog, horse, and human across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular class. The different TPR and FPR values are obtained at different levels of pruning. The blue circle marker denotes the values for bird, the red "*" marker denotes the values for dog, the magenta square marker denotes the values for horse, and the black star (pentagram) marker denotes the values for human. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line. We observe all our values are above the random-chance line.

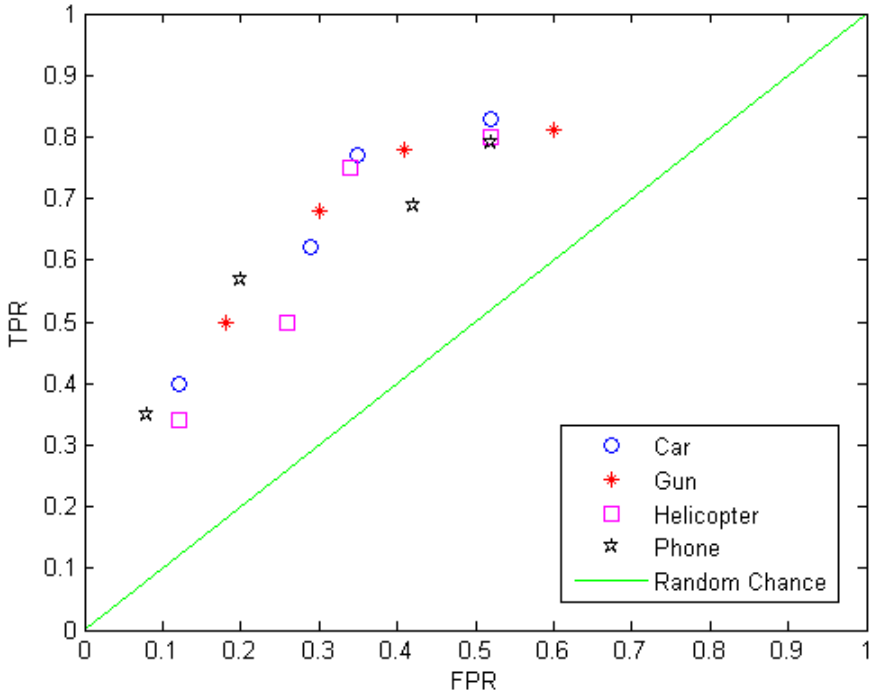


Figure 4.13: ROC plot for basic-inanimate level of classification using SVM. The x-axis denotes the false positive rate (FPR) and y-axis denotes the true positive rate (TPR). We average the TPR and FPR for each basic-inanimate category i.e., car, gun, helicopter, and phone across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular class. The different TPR and FPR values are obtained by varying the regularization parameter. The blue circle marker denotes the values for car, the red "*" marker denotes the values for gun, the magenta square marker denotes the values for helicopter, and the black star (pentagram) marker denotes the values for phone. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line. We observe all our values are above the random-chance line.

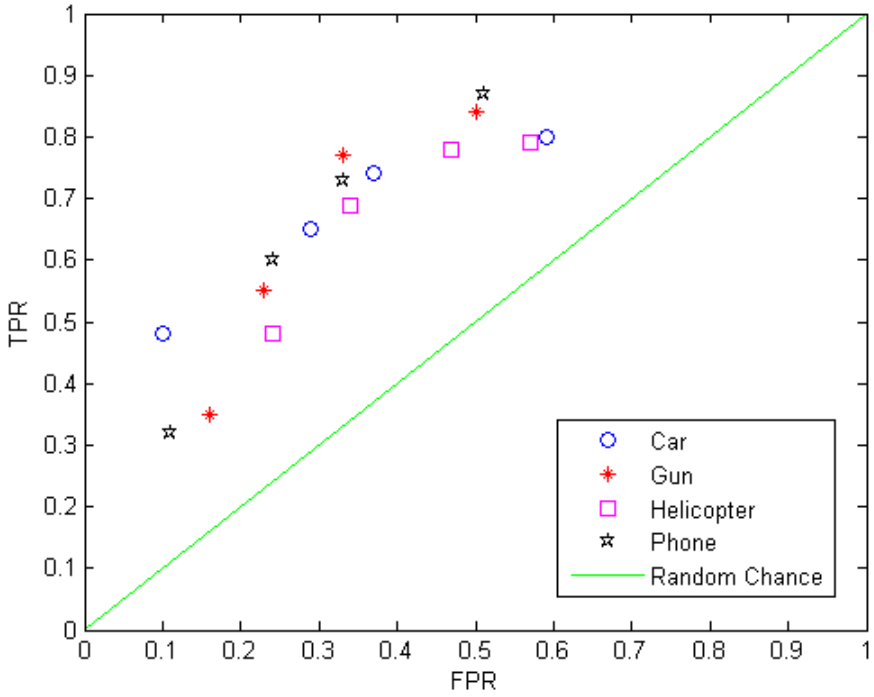


Figure 4.14: ROC plot for basic-inanimate level of classification using decision tree. The x-axis denotes the false positive rate (FPR) and y-axis denotes the true positive rate (TPR). We average the TPR and FPR for each basic-inanimate category i.e., car, gun, helicopter, and phone across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular class. The different TPR and FPR values are obtained by varying the different levels of pruning. The blue circle marker denotes the values for car, the red "*" marker denotes the values for gun, the magenta square marker denotes the values for helicopter, and the black star (pentagram) marker denotes the values for phone. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line. We observe all our values are above the random-chance line.

Figures 4.13 and 4.14 shows the ROC plots for basic-inanimate level of classification for SVM and decision tree respectively. We average the TPR and FPR for each basic-inanimate category i.e., car, gun, helicopter, and phone across all eight subjects. Each point in the graph shows the plot of FPR vs. TPR of a particular basic-inanimate category. The blue circle marker denotes the values for car, the red "*" marker denotes the values for gun, the magenta square marker denotes the values for helicopter, and the black star (pentagram) marker denotes the values for phone. The diagonal line connecting the bottom-left and the top-right points denote the random-chance line.

Figures 4.9, 4.11, and 4.13 are obtained by varying the regularization parameter of SVM [3], [78]. We notice that the TPR increases with the FPR. We get lower FPR values when we make the decision boundary very strict by increasing the regularization parameter. But at the same time the TPR values also drop. This result is due the small number of true positives when the boundary is very strict. We also get a few false negatives because of the strict boundary limit. But as we relax the boundary conditions (decrease the regularization parameter), we observe that TPR values increase. This result is due to the increase in the true positive values. We do get a few false positives as well, and hence FPR values also increase as we relax the decision boundary.

Figures 4.10, 4.12, and 4.14 are obtained varying the pruning criteria while building the decision tree [53]. We obtain lower FPR when the tree is not pruned. We notice that TPR values are also not very high when FPR values are low. This result is due to overfitting of the data caused as a result of the unpruned tree. As we prune the tree, we do get an increase in the true positive values, but we also see an increase in the false positives. We have to stop pruning at a certain threshold because pruning the tree a lot results in underfitting of data.

In order to improve our TPR values, we tried to vary the intercept parameter of SVM [3] and the impurity threshold for the decision tree. We did not obtain any improvements in the TPR values, however. So, we think that the TPR and FPR values depend on the data set. If the data has a high signal-to-noise ratio, we will obtain high TPR values and low FPR values.

Apart from accuracy, we also measured the time taken to perform a four-fold cross-validation on the data set using decision tree and SVM. SVM performs twice as fast when compared with decision tree. But, this result could be due to the fact that SVM and decision tree are implemented using different languages, Python and MATLAB respectively. We cannot say anything conclusive about the speed of the algorithms based on this result.

According to Kuncheva et al. decision tree combined with ensemble techniques produce much better results. We did not apply these techniques to our data set in this project. We can try these methods in the future, to determine whether ensemble techniques provide better accuracy.

Despite the fact that CART is not very popular among the neuroscience community, it is definitely an important algorithm. CART deals with missing feature values very efficiently, and hence the outcome is not affected. CART is also not affected by outliers. As we mentioned previously, CART effectively deals with high-dimensional data set, which will always be the case with fMRI data. It requires no prior feature selection techniques unlike SVM. Since decision tree performs feature selection as part of the classification, we can also use decision tree for feature selection i.e., select the top features obtained from decision tree, and apply SVM on these features.

Decision trees have a couple of drawbacks. The tree generated is always not the optimal tree since decision tree uses a greedy approach at each split in order to make local optimal decisions. Decision trees tend to overfit the data if a complete tree is

generated. Methods such as pruning helps to solve this problem. Pruning will require additional time for computation, however.

4.3 PyMVPA Programming Environment

We performed most of our analyses using the PyMVPA toolbox. This toolbox provides APIs to perform cross-validation, computing the dissimilarity matrices, and it also has an interface to store the result per voxel in a NifTI file. Furthermore, this toolbox interfaces with the libSVM module to provide an implementation of SVM algorithm for both two-class and multi-class classification. But the main drawback is that the users need to know Python programming to use this toolbox efficiently. To overcome this problem, we have come up with a programming environment that packages a set of commands that would enable the users to input different parameters, such as the data set, classifiers, dissimilarity matrix, and the output result file. Furthermore, the tool allows the users to customize their input. The tool itself is programmed in Python to provide the ease of using the PyMVPA toolbox. We have implemented the tool on Linux platform. When new features are added to PyMVPA, we can modify this tool easily by calling the relevant APIs. The tool is simple to use and we have compiled a manual for the tool that explains the usage with examples. Currently, we provide a command-line interface for the tool.

Example

Here is a simple example elucidating the usage of the tool:

python fmriCmd.py dataset *wrbold1A.nii.gz* samples *attrFile.txt* mask *mask.img* classifier *LinearCSVMC* select *[1,2,3,4,5,6,7,8]* crossvalidate *nfold 1* searchlight *11.0* accuracy

In the above command we provide the data set, mask file, and the attribute file as the input. We also specify the classifier as LinearCSVMC (a variation of linear SVM classifier.) The data will be de-trended and z-scored by default, and hence they are not mandatory inputs. The tool performs classification of the input data using the SVM classifier with searchlight of radius 11. In the example, we show leave-one-out cross validation. We select all the eight labels (specified as a Python list) with the select option. We specify accuracy as an option since searchlight returns error, and we are interested in computing the accuracy values. The accuracy values are stored per voxel, and the result is stored in a file that can be specified by the user. If the user does not specify a file name, the tool generates a file with a name "file" followed by the current time stamp and saves the result in a NifTI format.

Chapter 5

Related work and discussion

This section outlines a few related articles to this thesis. There are numerous studies in the past that have used MVPA for fMRI data analysis. Most of the studies so far have focused on analyzing the data obtained from visual stimuli. Haxby et al. used correlation measures as a means to distinguish different categories such as faces and houses. They analyzed a fMRI data obtained using visual stimuli [28]. These correlation values indicated that there are overlapping neural patterns that enable object categorization. MVPA was also used to predict the orientation and position of visual stimuli by Kamitani and Tong [30]. They used SVM on the data set with leave-one-out cross-validation technique to predict the orientation of the visual stimuli. A more recent study by Kuncheva et al. shows that classifier ensemble techniques performed better when compared with SVM [40]. They apply various machine-learning techniques on the popular Haxby data set. Yamashita et al. have shown an exhaustive comparison between SVM and sparse multinomial logistic regression (SMLR) classifiers [75]. They conclude that SMLR can remove irrelevant features, and thereby improves classifier performance. They use data set obtained by Kamitani and Tong [30] for their study. Davatzikos et al. apply machine-learning methods such as SVM to classify spatial patterns of brain activity for lie-detection methods [11]. They used visual cues such as

playing cards as stimuli.

MVPA can also be used for "mind-reading" techniques. We list a few of the studies in which the classifier tries to predict what the subject is responding to by looking at the brain activity. Haynes and Rees [23] predicted the stream of consciousness from brain activity. In their works, they outline different methods that can used to read the brain data and predict what the subject is thinking. Thirion et al. applied inverse retinotopy to reconstruct the patterns imagined by a subject just by reading the brain data [2]. They used SVM for classification, and obtained very high accuracy rate. A similar study was carried out by Shinkareva et al. in which the authors identify the cognitive states when the subject is responds to stimuli such as tools and dwellings [60]. Kay et al. used receptive field model with Pearson correlation to predict novel images from brain activity in early visual cortex [31].

Auditory data analysis using MVPA has gained interest recently. Staeren et al. showed that sound categories were characterized by overlapping neural responses in the auditory cortex. This result is similar to Haxby et al's result for visual data [62]. They presented the subjects with human, cat, and guitar sounds and studied the bilateral temporal lobe region for patterns. They found a large expanse of bilateral auditory areas were able to distinguish the sound categories. Formisano et al. showed different vowel sounds were represented via neural responses within the superior temporal lobes [17]. Ramirez et al. applied different machine algorithms to fMRI data for detecting instantaneous cognitive states [57]. Their data was obtained using melodies as stimuli. They compared the results obtained from various algorithms such as decision trees, SVM, artificial neural networks, and ensemble techniques. They use voxel discriminability and voxel activity for feature selection. A similar study was performed by Singh et al. where the authors apply GNB, SVM and k-NN to fMRI data, and try to uncover the cognitive states [61]. Raizada et al. [56] show that pattern information analysis can

detect patterns even when there is no change in regional-average activation. They validate their hypothesis by testing the responses to speech phonemes “ra” and “la” in the auditory cortex of native and non-native English speakers.

As we had mentioned previously, fMRI data set has large number of features. Hence, these data sets undergo a dimensionality reduction phase before applying machine-learning classifiers. In this passage, we outline a few of the feature selection methods, with relevant works. Tohka et al. applied independent component analysis (ICA) to fMRI data, that was later subjected to decision-tree classification with a Neyman-Pearson framework [70]. They conclude that performing ICA on the fMRI data significantly improves the quality of the data. Another technique is to use searchlight algorithm developed by Kriegeskorte et al. on the entire brain data [37]. Searchlight helps in selecting nearby voxels and computing the accuracy within the entire brain. By applying this technique, we can uncover all the brain areas activated by a particular stimuli. Natural scenes were categorized using searchlight algorithm with SVM [72] by Walther et al. They found significant differences in the representation of various natural scenes in the human visual cortex. ROI based analysis is another selection technique. Etzel et al. provide a nice analysis of ROI-based fMRI classification [15]. They use auditory sounds such as hand-action sounds (ripping of paper) as a stimuli to obtain fMRI data.

We also explore a few articles relating to hierarchical representation of the data within the human brain. Hierarchical division of the data set such as superordinate and basic categories has been studied since the 1980s. In visual categorization, the notion of hierarchy was hypothesized by Jolicoeur et al. This theory was based on the observation that the reaction time to basic level categories such as dog was a lot faster than superordinate level such as animal, and subordinate level such as terrier [27]. Studies conducted by Liu et al. using MEG, indicated faster responses to basic level when

compared with subordinate classes. The response time was higher when the subject performed object categorization task such as face vs. house when compared with the identification task such as identification of two faces [45]. Extending these results to the auditory data, Adams et al. tested categorization at basic and subordinate level for both auditory and visual data. The results indicated that the subjects performed more accurately at the basic levels when compared with the subordinate levels [1]. These studies also revealed areas in the frontal lobe participating in the auditory and visual categorization at subordinate levels, indicating finer levels of processing required involvement from other regions. Another conclusion that can be drawn from this result is, it also signifies that a particular category is processed by a smaller cortical loci. Another finding by Lewis et al. that a region in middle superior temporal gyrus is more activated by animate sounds when compared with inanimate sounds [29]. This results shows there is category-specific preference in the auditory areas. These results are in stark contrast to the recent findings by Staeren et al. who claim that there are no such category-specific modules within the temporal lobes [62]. Our data set has two-levels of hierarchy associated with it, and hence we explore encoding of data in the brain at these different levels.

MVPA analysis is not limited to classification alone. Representational similarity analysis, a technique developed by Kriegeskorte et al. is also a popular method to analyze the fMRI data [38]. Edelman et al. were the pioneers to apply the similarity analysis to fMRI data, and they represented these similarities using multidimensional scaling [13]. They analyzed data obtained from seven subjects using cars, planes, animals, and fish as the stimuli. Based on Shepard's theory of second-order isomorphism, Edelmen [14] proposed a method for shape representation to establish a relationship between internal representation space to that of the world objects. He uses concepts such as distal and proximal similarities, and learning representations from examples to

strengthen his theory. Laakso et al. have shown that correlation between inter-point distances in any two activation spaces measures the representational similarity [41]. They proved that individuals with different number of neurons represent the world in similar ways. Beeck et al. used the data obtained from macaque's inferotemporal cortex to study the similarity of different shapes [12]. Kayaert et al. also studied macaque's inferotemporal cortex for representation of regular and irregular shapes. They project these similarities using MDS [32]. Previously mentioned research such as Haxby et al. [28], Kay et al. [31], and Shinkareva et al. [60] use similarity analysis in their study. Similarity analysis involves averaging of neighboring voxels for robust correlation estimation. Rydell et al. have proposed a method to average the voxels with similar correlation coefficients [58]. According to the authors, this methods produces better separability between different voxels. Kriegeskorte et al. used RSA to analyze the similarity patterns in the inferior temporal areas of man and monkey [39]. More recently, Clithero et al. performed cross-subject analysis by using RSA combined with feature-selection methods [9]. They used data obtained from visual stimuli when subjects were viewing faces. Graham et al. perform similarity analysis on art data [19]. They mainly use MDS to project their results for different portrait and landscape paintings (Refer to Appendix A for more details.) The previous studies have emphasized on the similarity analysis of data obtained from visual area. There are not many similarity studies that involve auditory stimuli. Recently, Mahon et al. performed similarity analysis on the data obtained by using auditory words as input [46]. They study the conceptual distance between pairs of auditory words such as "chairs" and "stools" that are more similar to each other, and "chairs" and "stove" that more dissimilar to each other.

MDS is one form of representing the similarity. There are also other techniques like dendrograms and clustering algorithms that can be used to represent the similarity structure. Discovery of structure in a data has been around for centuries. Carl Linnaeus

organized the biological species in a hierarchy. Mendeleev grouped the elements in a periodic table. In a similar manner, most of the data have some form structure associated with it. For example, social networks can be represented as cliques. Kemp et al. have found an approach using graph grammars and hierarchical clustering to fit a structure to relational data, similarity data, and feature data [34]. The authors propose that their algorithm uncovers the optimal structure of representing the data. In this paragraph and next paragraph, we outline a few researches that are related to Kemp's algorithm. Inhelder et al. explained the development of logic in a child [26]. The authors say that initially a child represents the data in a simple structure; but when the child learns new data, this structural form becomes more complicated, and similar things get clustered together, and the child tries to generalize the information by such groups.

The concept of clustering and tree-fitting using graph grammars was proposed by Roger Shepard in 1980 [59]. Shepard showed that by applying simple generative process on graph grammars, complex forms such as trees and chains can be created. There are mainly two forms of clustering algorithms such as parametric clustering and non-parametric clustering. Parametric approach includes generative models such as Gaussian mixture model (GMM) and reconstructive models such as k-means. Non-parametric approach includes agglomerative and divisive (hierarchical) clustering algorithms. There are many hierarchical clustering algorithms such as average linkage clustering, centroid linkage clustering, complete linkage clustering, single linkage clustering, and Ward's method. Studies such as [18] and [63] provide a comprehensive overview of these algorithms with different examples. Kim et al. provide a method to weigh the links in a structural form called "LinkRank" [35]. The authors say that this ranking method is useful for locating communities within networks. Braun et al. propose a method for structure learning using Bayesian principles [6]. Fiser et al. use maximum a posteriori probability (MAP) estimate for developing an optimal learning

model in humans [16]. They propose a method for representing information and uncertainty in the cortex. Brady et al. show that humans tend to remember objects that have some similar patterns in them [5]. They presented the subjects with similar colors and dissimilar color patterns. They observed that humans tended to remember the similar color patterns more distinctively. The authors claim that the objects represented using some form of grouping are stored in a compressed manner in the memory due to data redundancy. They say that the compression helps in efficient coding of data in the brain. We are not aware of any fMRI study that uses Kemp's algorithm for similarity analysis. We are the first to apply Kemp's algorithm to an fMRI data set.

Numerous software are available for performing MVPA. We mainly use a software developed by Hanke et al. called PyMVPA [22] for our analysis. We list here a few of the researches who use PyMVPA in their study. As we mentioned earlier, Clithero et al. used PyMVPA for performing similarity analysis across subjects [9]. Sun et al. used SMLR classifier in PyMPVA to analyze the fMRI data obtained from patients having recent onset of psychosis [66]. They obtained 86.1% accuracy using this classifier with leave-one-out cross-validation. They discovered that there was a significant deficiency of gray matter in the prefrontal, cingulate, and lateral temporal regions in the patients when compared with the healthy subjects. Hence, the patterns obtained from the recent-onset psychotic subjects can be used to determine the risk of developing psychosis or schizophrenia. Wang et al. used conformal slit mapping, tensor-based morphometry, and a classifier provided by PyMVPA called multinomial logistic regression classifier to identify cortical biomarkers for classification [74], [73]. They obtained 92.7% accuracy for their data set that had patients suffering from Williams syndrome. PyMVPA is just not limited to fMRI data, but also to other kinds of data set. Trautmann et al. designed a mobile robot for penetrating radar surveys of polar ice [71]. They used SVM in PyMVPA for autonomous rough terrain negotiation i.e., to detect immobilization and

hazards. They obtained 98.4% accuracy with their test data set.

The previous studies and research motivated us to go a step further to unify various numerical techniques together and apply it on a data set. The previous researches mostly use SVM for classification, and hence we feel that it is apt to test our data set with the method as well. Decision tree is not that popular with the neuroscience community, but we analyzed our data set with it to find out the difference between a hierarchical vs. a non-hierarchical algorithm. Our data set is also unique because it has different levels of representation, and it is created using auditory stimuli. Since auditory area is not very widely explored in the field of neuroimaging, we illustrate our methods with an auditory data set. We also find a novel structural form of our data using an algorithm developed by Kemp et al. This information provides us an insight as to how our data set could be represented in the human brain.

Classification results using SVM and searchlight yielded areas in the brain that respond to auditory stimuli. Most of the voxels are found in the superior and middle temporal regions of the brain. We uncovered a few voxels in the occipital and frontal lobes as well. We compared the results of SVM with decision-tree algorithm. Both algorithms produce similar accuracy values for our data set.

We computed RDMs using the RSA technique on the entire brain data and did not find any striking patterns within the data. We generated RDMs for various ROIs, and found that these matrices are represented more clearly using MDS. We found out that "ring" and "tree" are voted the best forms for our data set, when we applied Kemp's algorithm on our data set.

We have developed a programming environment suite for PyMVPA that would enable the neuroscience community to effectively perform data analysis by executing a few commands without having to learn the Python programming language.

The main goal of this paper is to unify different numerical methods and illustrate with an example how to use them for data analysis. Hence the studies presented in this paper would help in better understanding of the different numerical methods that can be applied to fMRI data. The programming environment makes the computational tools required for data analysis more accessible to the neuroscientists. Since it requires execution of only a few commands to run the analysis, it is a huge time saver. We believe that more users would be using our tool since it avoids learning a programming language. We hope that the study of the sample data set would reveal interesting information about how the auditory engines of human brains decode various stimuli.

We have published the source code of our implementation with documentation and provide references to the other software used in this research, so that they can be easily accessed by the neuroscientists. Please refer to Appendix B for more details.

Some of the extensions to this project are as follows:

- Porting the decision-tree algorithm code to PyMVPA.
- Developing a graphical interface to the command-line PyMVPA wrapper module.
- Developing the MDS algorithm for PyMVPA.
- Porting the Kemp's algorithm to PyMVPA.

Appendix A

Dissimilarity Analysis

We generated RDMs for the data obtained from entire brain and also various ROIs, and we analyzed them using Kemp's algorithm. In this section, we show the dissimilarity matrices obtained for the data obtained from the entire brain, middle temporal regions, and inferior occipital regions.

Figures A.1, A.2, and A.3 show the dissimilarity matrix of the eight basic categories for the entire brain, middle temporal, and inferior occipital areas respectively. The rows and columns indicate the eight basic categories such as bird, dog, ... The matrices are obtained after averaging the RDMs across all the eight subjects. The colors indicate the correlation $1 - r$, r being the Spearman correlation. Blue indicates strong correlation among the data, where as red indicates weak correlation.

From Figure A.1 we observe that all the correlation values are about 0.6. We see no significant patterns to conclude the similarity between the different categories. We think this results could be due to the fact the the data from the entire brain is noisy (Please refer to Section 4.1 for more details.)

Figure A.2 shows that bird and dog sounds have some similarity between them. The next best similarity values are obtained for dog and horse sounds. The other categories do not exhibit good correlation patterns. They have correlation values closer to or

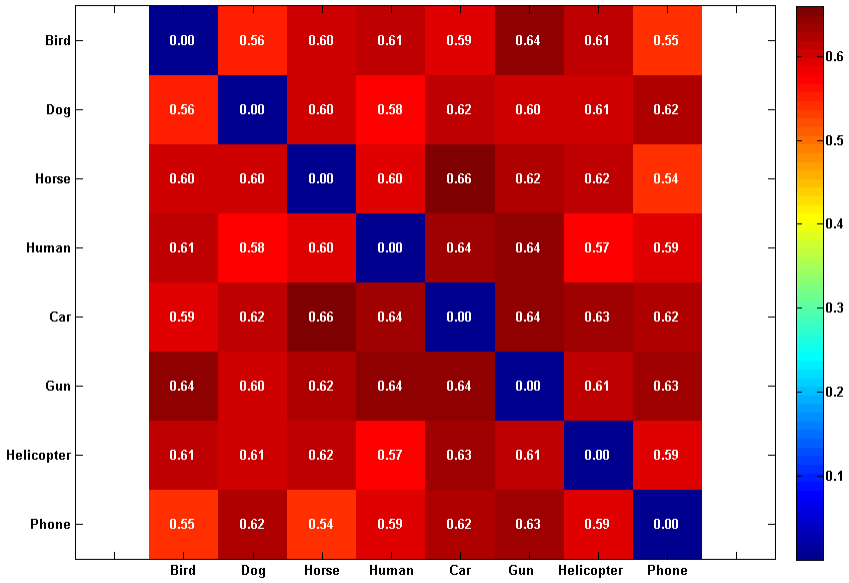


Figure A.1: The dissimilarity matrix of the eight basic categories for the entire brain data. The matrix is symmetric about the diagonal. The rows and columns indicate the eight basic categories such as bird, dog, ... The matrix is obtained after averaging the RDMs obtained using the entire brain data, across all eight subjects. The colors indicate the correlation $1 - r$, r being the Spearman correlation. Blue indicates strong correlation among the data, where as red indicates weak correlation. We did not find any striking dissimilarities within the data set for the entire brain.

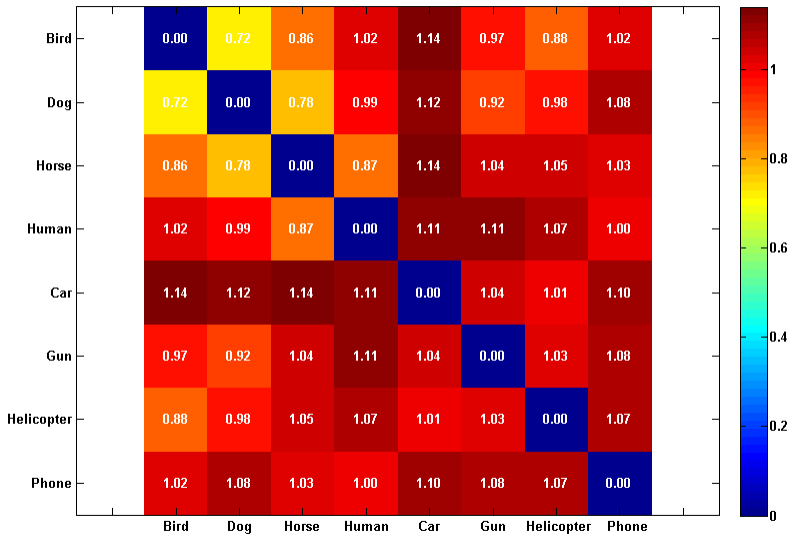


Figure A.2: Dissimilarity matrix of the eight basic categories for the middle temporal area. The matrix is symmetric about the diagonal. The rows and columns indicate the eight basic categories such as bird, dog, ... The matrix is obtained after averaging the RDMs obtained using the middle temporal data across all the eight subjects. The colors indicate the correlation $1 - r$, r being the Spearman correlation. Blue indicates strong correlation among the data, where as red indicates weak correlation.

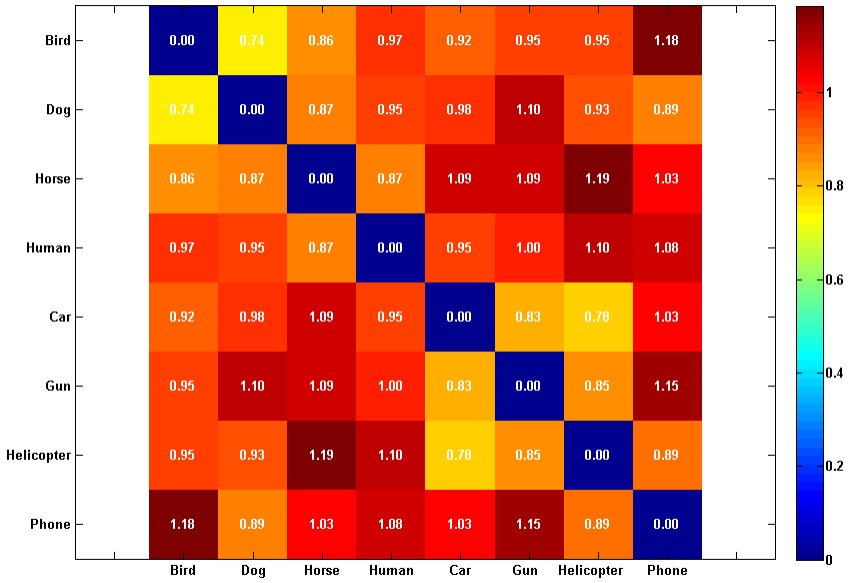


Figure A.3: Dissimilarity matrix of the eight basic categories for the inferior occipital area. The matrix is symmetric about the diagonal. The rows and columns indicate the eight basic categories such as bird, dog, ... The matrix is obtained after averaging the RDMs obtained using the inferior occipital data across all the eight subjects. The colors indicate the correlation $1 - r$, r being the Spearman correlation. Blue indicates strong correlation among the data, where as red indicates weak correlation.

greater than 1. From this representation we think that middle temporal area may be tuned to code for animate categories (Please refer to Section 4.1 for more details.)

Figure A.3 shows that bird and dog sounds have some similarity between them. The next best similarity values are obtained for car and helicopter sounds. The other categories such as car and gun exhibit some similarity as well. The other categories have correlation values closer to or greater than 1. Phone is not correlated with any of the animate or inanimate sounds.

From Figures A.2 and A.3, we can observe a little better correlation pattern than the entire brain data. But, we feel other forms of representations may help in visualizing the similarity measures more clearly. Hence, we project these matrices using MDS and also apply Kemp's algorithm on this data (Please refer to Section 4.1 for more details.) We feel that MDS and Kemp's result provide a better representation for the similarity analysis.

We analyzed Connolly et al's [10] data set with Kemp's algorithm and obtained "Ring" and "Tree" as the representational form for this data set as well. We observe that "Ring" and "Tree" are the most voted pattern of representation for data sets that have some form of grouping among the different categories.

In order to check whether Kemp's algorithm is applicable for other kinds of data, we analyzed a dissimilarity matrix obtained using art data. The data consisted of different portraits of people, fruits, and flowers.[1] The dissimilarity matrix was obtained by averaging the dissimilarity values across 19 subjects.

The dissimilarity matrices were provided by Daniel J. Graham [19]. We obtain "Tree" as the best representational structural form for this data set.

Figure A.4 shows the two dimensional MDS representation for the portrait art data.[2]

[1] Please refer to [19] for the description of the various paintings used in this study.

[2] The MDS representation has been reproduced from [19].

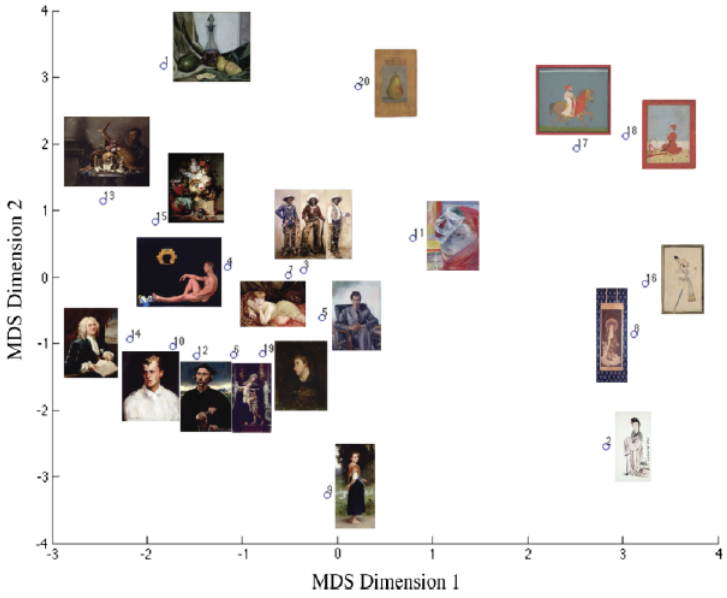


Figure A.4: Two dimensional MDS structure for the portrait paintings. MDS was computed over the average dissimilarity values obtained from 19 subjects. In this structure, we can see that the Asian paintings are clustered together, men and women form another group, fruits and flowers are placed near each other.

There were 20 paintings and 19 subjects in this experiment. MDS was computed over the average dissimilarity values obtained from 19 subjects. In this structure, we can see that the Asian paintings (Sultans and Japanese) are clustered together, men and women form another group, fruits and flowers are placed near each other.

Figure A.5 shows how the different portrait paintings are represented using the structural form algorithm. We used the dissimilarity matrix obtained by averaging the matrices of all the 19 subjects as input to Kemp's algorithm. We obtain "tree" as the best voted representation for this data set. "Ring" was chosen the second best representation for this data. In this structure, we can see that the Asian paintings (sultans and Japanese paintings) are clustered together, fruits and flowers form another cluster, the girls are clustered together, and all the men form another cluster. We observe that all the Sultans form a subcluster, the Japanese art form another subcluster within the Asian paintings. The paintings labeled "man" and "lady" in the figure are clustered

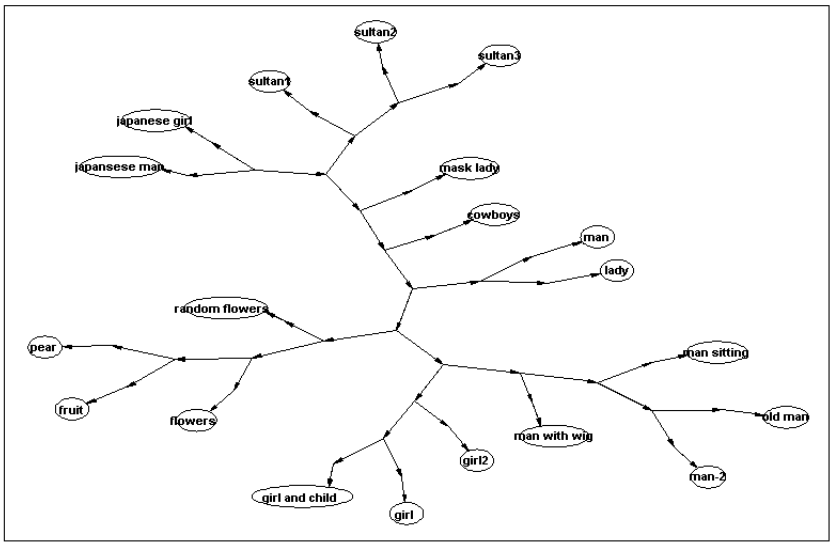


Figure A.5: Structural form for the dissimilarity matrix obtained from art data. We obtained a tree structure for this data using Kemp's algorithm. In this structure, we can see that the Asian paintings (sultans and Japanese paintings) are clustered together, fruits and flowers form another cluster, the girls are clustered together, and all the men form another cluster.

together because they have similar appearance. The painting labeled “mask lady” does not fall in the category of “girl” because the painting of “mask lady” is not very clear to observe. The painting labeled “cowboys” is separate from the other men category since they do not have similar appearance. Furthermore, we analyzed each individual subjects dissimilarity matrix and obtained “tree” as the structural form for 18 subjects. One subject’s matrix generated the “ring” structure.

If we observe the dissimilarity matrix itself (not a part of this paper), we can see that MDS and Kemp’s algorithm produce a much better representation of the similarity values of different categories.

Another useful representation for the dissimilarity matrices is the dendrogram [48]. In this representation, the dissimilarity matrix is represented in the form of a hierarchical tree.

Appendix B

Source Code

The source code for the PyMVPA wrapper tool is located on the Dartmouth College Computer Science Server. The URL to the webpage is as follows:

`http://www.cs.dartmouth.edu/~geethmala/Research.html`.

The source code and scripts are located in a folder called "wrapper". We also provide the links to the external resources we used for this project in the "README.txt" file. The manual for the PyMVPA wrapper tool is located inside the "wrapper" folder called "manuals". We provide a script to convert PyMVPA ".dat" files to MATLAB ".mat" files. This script will be useful when the users want to run Kemp's algorithm on MATLAB using the RDMs obtained from PyMVPA. This script can be found in the folder "misc".

Please refer to the "README.txt" file on the webpage for more details.

Bibliography

[1] Reginald B. Adams and Petr Janata. A comparison of neural circuits underlying auditory and visual object categorization. *NeuroImage*, 16(2):361 – 377, 2002.

[2] Thirion B., Duchesnay E., Hubbard E., Dubois J., Poline J.B., Lebihan D., and Dehaene S. Inverse retinotopy: Inferring the visual content of images from brain activation patterns. *Neuroimage*, 33(4):1104–1116, 2006.

[3] Francis R. Bach, David Heckerman, and Eric Horvitz. On the path to an ideal ROC curve: Considering cost asymmetry in learning classifiers. In *Proceedings of the Tenth International Workshop on Artificial Intelligence and Statistics (AISTATS)*, pages 9–16, 2005.

[4] Paul Boersma and David Weenink. Praat: doing phonetics by computer (version 5.1.05) [computer program], 2009. http://www.fon.hum.uva.nl/praat/.

[5] Timothy F. Brady, Talia Konkle, and George A. Alvarez. Compression in visual working memory: Using statistical regularities to form more efficient memory representations. *Journal of Experimental Psychology: General*, 138(4):487 – 502, 2009.

[6] Daniel A. Braun, Carsten Mehring, and Daniel M. Wolpert. Structure learning in action. *Behavioural Brain Research*, 206(2):157 – 165, 2010.

[7] Leo Breiman, J. H. Friedman, R. A. Olshen, and C. J. Stone. *Classification and Regression Trees*. Wadsworth, 1984.

[8] Matthew Brett, J.-L. Anton, R. Valabregue, and J.-B. Poline. Region of interest analysis using an SPM toolbox. *NeuroImage*, 16(2), June 2002. Presented at the 8th International Conference on Functional Mapping of the Human Brain, June 2–6, 2002, Sendai, Japan. Available on CD-ROM.

[9] John A. Clithero, David V. Smith, R. McKell Carter, and Scott A. Huettel. Within- and cross-participant classifiers reveal different neural coding of information. *NeuroImage*, In Press, Corrected Proof:–, 2010.

[10] Andrew C. Connolly, M. Ida Gobbini, and James V. Haxby. Three virtues of similarity-based multivoxel pattern analysis: An example from the human object vision pathway.

[11] C. Davatzikos, K. Ruparel, Y. Fan, D.G. Shen, M. Acharyya, J.W. Loughead, R.C. Gur, and D.D. Langleben. Classifying spatial patterns of brain activity with machine learning methods: Application to lie detection. *NeuroImage*, 28(3):663–668, 2005.

[12] Hans Op de Beeck, Johan Wagemans, and Rufin Vogels. Inferotemporal neurons represent low-dimensional configurations of parameterized shapes. *Nature Neuroscience*, 4:1244–1252, 2001.

[13] S. Edelman, K. Grill-Spector, T. Kushnir, and R. Malach. Towards direct visualization of the internal shape space by fMRI. *Psychobiology*, 26:309–321, 1998.

[14] Shimon Edelman. Representation is representation of similarities. *Behavioral and Brain Sciences*, 21(04):449–467, 1998.

[15] Joset A. Etzel, Valeria Gazzola, and Christian Keysers. An introduction to anatomical ROI-based fMRI classification analysis. *Brain Research*, 1282:114–125, 2009.

[16] József Fiser, Pietro Berkes, Gergő Orbán, and Máté Lengyel. Statistically optimal perception and learning: from behavior to neural representations. *Trends in Cognitive Sciences*, 14(3):119 – 130, 2010.

[17] Elia Formisano, Federico De Martino, Milene Bonte, and Rainer Goebel. "Who" is saying "What"? Brain-based decoding of human voice and speech. *Science*, 322(5903):970–973, 2008.

[18] Glenn Fung. A comprehensive overview of basic clustering algorithms.

[19] Daniel J. Graham, Jay D. Friedenberg, Daniel N. Rockmore, and David J. Field. Mapping the similarity space of paintings: Image statistics and visual perception. *Visual Cognition*, 18(4):559–573, 2010.

[20] Data Format Working Group. The Nifti-1 Data Format. `http://nifti.nimh.nih.gov/nifti-1`.

[21] Michael Hanke and Yaroslav O. Halchenko. PyMVPA overview. `http://pymvpa.org/overview.html`.

[22] Michael Hanke, Yaroslav O. Halchenko, Per B. Sederberg, Stephen Jose Hanson, James V. Haxby, and Stefan Pollmann. PyMVPA: A Python toolbox for multivariate pattern analysis of fMRI data. *Neuroinformatics*, 7(1):37–53, 2009.

[23] John-Dylan Haynes and Geraint Rees. Predicting the stream of consciousness from activity in human visual cortex. *Current Biology*, 15(14):1301–1307, 2005.

[24] IBM. Decision tree cart algorithm.

[25] IBM. IBM SPSS Software. http://www.spss.com/software/statistics/.

[26] Barbel Inhelder and Jean Piaget. Classification and seriation. In *The Early Growth of Logic in the Child*, page 302. W. W. Norton & Company, Inc, 1969.

[27] P. Jolicoeur, M. A. Gluck, and S. M Kossyln. Pictures and names: making the connection. *Cognitive Psychology*, 16(2):243–275, 1984.

[28] Haxby JV, Gobbini M, Furey ML, Ishal A, Schouten JL, and Pietrini P. Distributed and overlapping representations of faces and objects in ventral temporal cortex. *Science*, 293:2425–2430, 2001.

[29] Lewis J.W., Brefczynski J.A., Phinney R.E., Janik J.J., and DeYoe E.A. Distinct cortical pathways for processing tool versus animal sounds. *Nature Neuroscience*, 25(21):910–916, 2005.

[30] Yukiyasu Kamitani and Frank Tong. Decoding the visual and subjective contents of the human brain. *Nature Neuroscience*, 8(5):679–685, 2005.

[31] Kendrick N. Kay, Thomas Naselaris, Ryan J. Prenger, and Jack L. Gallant. Identifying natural images from human brain activity. *Nature*, 452(7185):352–355, 2008.

[32] Greet Kayaert, Irving Biederman, and Rufin Vogels. Representation of regular and irregular shapes in macaque inferotemporal cortex. *Cerebral Cortex*, 15:1308–1321, 2005.

[33] Charles Kemp and Joshua B. Tenenbaum. Form discovery software. `http://www.psy.cmu.edu/~ckemp/code/formdiscovery.html`.

[34] Charles Kemp and Joshua B. Tenenbaum. The discovery of structural form. *Proceedings of the National Academy of Sciences*, 105(31):10687–10692, 2008.

[35] Youngdo Kim, Seung-Woo Son, and Hawoong Jeong. Finding communities in directed networks. *Phys. Rev. E*, 81(1):016103, Jan 2010.

[36] S.M. Kosslyn, G. Ganis, and W.L. Thompson. Neural foundations of imagery. *Nature Reviews Neuroscience*, 2(9):635–642, 2001.

[37] Nikolaus Kriegeskorte, Rainer Goebel, and Peter Bandettini. Information-based functional brain mapping. *Proceedings of the National Academy of Sciences of the United States of America*, 103(10):3863–3868, 2006.

[38] Nikolaus Kriegeskorte, Marieke Mur, and Peter Bandettini. Representational similarity analysis—connecting the branches of systems neuroscience. *Frontiers in System Neuroscience*, 2(4), 2008.

[39] Nikolaus Kriegeskorte, Marieke Mur, Douglas A. Ruff, Roozbeh Kiani, Jerzy Bodurka, Hossein Esteky, Keiji Tanaka, and Peter A. Bandettini. Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron*, 60(6):1126–1141, 2008.

[40] Ludmila I. Kuncheva and Juan J. Rodríguez. Classifier ensembles for fMRI data analysis: an experiment. *Magnetic resonance imaging*, 28(4):583–593, 2010.

[41] Aarre Laakso, , Aarre Laakso, and Garrison Cottrell. Content and cluster analysis: Assessing representational similarity in neural systems. *Philosophical Psychology*, 13:47–76, 2000.

[42] Sverine Lambert, Eliana Sampaio, Christian Scheiber, and Yves Mauss. Neural substrates of animal mental imagery: calcarine sulcus and dorsal pathway involvement–an fMRI study. *Brain Research*, 924(2):176 – 183, 2002.

[43] Nicole A. Lazar, Beatriz Luna, John A. Sweeney, and William F. Eddy. Combining brains: A survey of methods for statistical pooling of information. *NeuroImage*, 16(2):538–550, 2002.

[44] Yune-Sang Lee. *Neural basis underlying auditory categorization in the human brain*. PhD thesis, Dartmouth College, 2010.

[45] Jia Liu, Alison Harris, and Nancy Kanwisher. Stages of processing in face perception: an MEG study. *Nature Neuroscience*, 5(9):910–916, 2002.

[46] B.Z. Mahon and A. Caramazza. Judging semantic similarity: An event-related fMRI study with auditory word stimuli. *Neuroscience*, 169:279–286, 2010.

[47] The Mathworks. Decision tree (classification and regression tree) implementation. http://www.mathworks.cn/access/helpdesk/help/toolbox/stats/classregtree.html.

[48] The Mathworks. Dendrogram plots.

[49] FIL methods group. Statistical Parametric Mapping. `http://www.fil.ion.ucl.ac.uk/spm`.

[50] Christian Müller. Fundamentals, features, and methods. In *Speaker Classification I*, page 355. Springer, 2007.

[51] Marieke Mur, Peter A. Bandettini, and Nikolaus Kriegeskorte. Revealing representational content with pattern-information fMRI—an introductory guide. *Social cognitive and affective neuroscience*, 4(1):101–109, 2009.

[52] Kenneth A. Norman, Sean M. Polyn, Greg J. Detre, and James V. Haxby. Beyond mind-reading: multi-voxel pattern analysis of fMRI data. *Trends in Cognitive Sciences*, 10(9):424–430, 2006.

[53] Andrea Padoan. Decision trees and predictive models with cross-validation and ROC analysis plot. http://www.mathworks.com/matlabcentral/fileexchange/26326.

[54] W.D. Penny, A.P. Holmes, and K.J. Friston. Random effects analysis. In R.S.J. Frackowiak, K.J. Friston, C. Frith, R. Dolan, K.J. Friston, C.J. Price, S. Zeki, J. Ashburner, and W.D. Penny, editors, *Human Brain Function*. Academic Press, 2nd edition, 2003.

[55] Francisco Pereira, Tom Mitchell, and Matthew Botvinick. Machine learning classifiers and fMRI: A tutorial overview. *NeuroImage*, 45(1, Supplement 1):S199–S209, 2009.

[56] Rajeev D. S. Raizada, Feng-Ming Tsao, Huei-Mei Liu, and Patricia K. Kuhl. Quantifying the adequacy of neural representations for a cross-language phonetic discrimination task: Prediction of individual differences. *Cerebral Cortex*, 20(1):1–12, 2009.

[57] Rafael Ramirez and Montserrat Puiggros. A machine learning approach to detecting instantaneous cognitive states from fMRI data. In *Advances in Knowledge Discovery and Data Mining*, pages 248–259. Springer Berlin / Heidelberg, 2007.

[58] J. Rydell, H. Knutsson, and M. Borga. Correlation controlled adaptive filtering for fMRI data analysis. In *In Proceedings of the 13th Nordic-Baltic conference on biomedical engineering and medical physics (NBC05)*, 2005.

[59] Roger N. Shepard. Multidimensional scaling, tree-fitting, and clustering, 1980.

[60] Svetlana V. Shinkareva, Robert A. Mason, Vicente L. Malave, Wei Wang, Tom M. Mitchell, and Marcel Adam Just. Using fMRI brain activation to identify cognitive states associated with perception of tools and dwellings. *PLoS ONE*, 3:e1394, 01 2008.

[61] Vishwajeet Singh, K. P. Miyapuram, and Raju S. Bapi. Detection of cognitive states from fMRI data using machine learning techniques. Technical report, In: Proceedings of Twentieth International Conference on Artificial Intelligence. (2007) 587–592, 2005.

[62] Nol Staeren, Hanna Renvall, Federico De Martino, Rainer Goebel, and Elia Formisano. Sound categories are represented as distributed patterns in the human auditory cortex. *Current Biology*, 19(6):498–502, 2009.

[63] StatSoft. A tutorial on cluster analysis. `http://www.statsoft.com/textbook/cluster-analysis/`.

[64] StatSoft. A tutorial on multidimensional scaling. `http://www.statsoft.com/textbook/multidimensional-scaling/`.

[65] S. C. Strother. Evaluating fMRI preprocessing pipelines. *IEEE Engineering in Medicine and Biology Magazine*, 25(2):27–41, 2006.

[66] Daqiang Sun, Theo G.M. van Erp, Paul M. Thompson, Carrie E. Bearden, Melita Daley, Leila Kushan, Molly E. Hardt, Keith H. Nuechterlein, Arthur W. Toga, and Tyrone D. Cannon. Elucidating a magnetic resonance imaging-based neuroanatomic biomarker for psychosis: Classification analysis using probabilistic brain atlas and machine learning algorithms. *Biological Psychiatry*, 66(11):1055 – 1060, 2009. Genotypic and Neuroimaging Biomarkers for Schizophrenia.

[67] John A. Swets. *Signal Detection Theory and Roc Analysis in Psychology and Diagnostics: Collected Papers*. Lawrence Erlbaum Associates, 1996.

[68] Kardi Teknomo. Tutorial on Decision Tree, 2009. `http://people.revoledu.com/kardi/tutorial/decisiontree`.

[69] Roman Timofeev. Classification and regression trees (CART) theory and applications. Master's thesis, Humboldt University, Berlin, 2004.

[70] Jussi Tohka, Karin Foerde, Adam R. Aron, Sabrina M. Tom, Arthur W. Toga, and Russell A. Poldrack. Automatic independent component labeling for artifact removal in fMRI. *NeuroImage*, 39(3):1227–1245, 2008.

[71] E. Trautmann, L. Ray, and J. Lever. Development of an autonomous robot for ground penetrating radar surveys of polar ice. In *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*, pages 1685 –1690, 10-15 2009.

[72] Dirk B. Walther, Eamon Caddigan, Li Fei-Fei, and Diane M. Beck. Natural scene categories revealed in distributed patterns of activity in the human brain. *Journal of Neuroscience*, 29(34):10573–10581, 2009.

[73] Yalin Wang, Tony F. Chan, Paul M. Thompson, and Shing-tung Yau. Conformal slit mapping and its applications to brain surface parameterization.

[74] Yalin Wang, Rudy Senstad, Arthur W. Toga, and Paul M. Thompson. MRI-based biomarker detection using conformal slit maps and machine learning.

[75] Okito Yamashita, Masa aki Sato, Taku Yoshioka, Frank Tong, and Yukiyasu Kamitani. Sparse estimation automatically selects voxels relevant for the decoding of fMRI activity patterns. *NeuroImage*, 42(4):1414–1429, 2008.

[76] Y. Yohannes and J. Hoddinott. Classification and regression trees:an introduction.

[77] E. Yumoto, W.J. Gould, and T. Baer. Harmonics-to-noise ratio as an index of the degree of hoarseness. *Journal of Speech and Hearing Research*, 27:2–6, 1984.

[78] Shaoyi Zhang, M. Maruf Hossain, Md. Rafiul Hassan, James Bailey, and Kotagiri Ramamohanarao. Feature weighted SVMs using receiver operating characteris-

tics. In *Proceedings of the 2009 SIAM International Conference on Data Mining (SDM09)*, pages 497–508, 2009.