CS 591 Submission 06:

Decoding fMRI for Modality, Stimulus, and Task Using SVC

Nishmar Cestero

Boston University

I. Data Collection

Data Origin

The dataset comes from the Perceptual Neuroimaging Lab at Boston University. It is composed of fMRI data of participants performing the Frontal Localizer task collected by Dr. Abigail Noyce.

During the experiment, participants were presented with a sequence of stimuli and were either asked to perform a Frontal Localizer task or passively attend to the stimuli. The Frontal Localizer task exercises working memory for the stimulus type presented. The participant was asked to indicate when the current stimulus matched the stimulus from two steps earlier in the sequence. Stimuli were either auditory or visual. Auditory stimuli were cat and dog vocalizations; visual stimuli were male and female faces.

There are 8 participants from this study in the dataset. Each participant performed in 8 runs of the experiment. Each stimulus sequencce was shown twice: once in the passive condition and once in the active condition. In addition, part of the data is composed of information that is not of interest: data from when the participant was being instructed or performing visual fixation.

fMRI Data

Functional Magnetic Resonance Imaging (fMRI) measures brain activity through changes in blood flow. The blood-oxygen-level dependent (BOLD) imaging technique used in fMRI provides an indirect measure of brain activity. It observes hemodynamic responses in neurons, which is the rapid delivery of blood to neuronal tissues. The more active a neuron is, the more oxygen it requires, so more blood must travel to the neuron to provide oxygen.

The BOLD signal is stored as a 3-dimensional image in voxels. Voxels are units representing a cube of brain tissue. Each voxel can represent up to a million neurons. The unit of measurement for voxels in the data collected is millimeters.

Retrieving and Preprocessing Data

Before Python can import and read fMRI data files, a standard set of steps must be performed to retrieve, preprocess, and get a preliminary analysis of the data. This process will output files that Python can interpret.

FreeSurfer Steps:

FreeSurfer is software that provides functions for processing and analyzing brain MRI images.

1. Unpack raw data from DICOM files

   DICOM stands for 'The Digital Imaging and Communications in Medicine.' It is the standard type for any kind of medical image.

2. Add paradigm files

   Each session consists of a predetermined amount of time with a schedule for stimuli the subject is presented with or activity the subject is asked to perform in the scanner. Paradigm files are text files containing stimuli information. They are manually copied to the run folder.

3. Preprocess

   Achieve motion correction, smoothing, projection to an average brain, and other necessary functions for noise reduction and proper image analysis. We use a file projected on an "average" brain, rather than the subject's own brain, in hopes of generalizing the classifier to more than one subject.

II. Nature of Attributes

Types

NIfTI Files

NIfTI refers to 'Neuroimaging Informatics Technology Initiative.' This is a standard file format for neuroimaging data, such as fMRI. After following the steps above with FreeSurfer, we get a NIfTI file, which we can access in Python through the NiBabel module.

In order to explore the attributes of NIfTI files in Python, I loaded a file for the left hemispheres of the brain projected on an "average" brain for subject 140930NP in run 01 into a NiBabel image.

NumPy Arrays

The NiBabel image can be converted to a NumPy array. The NumPy array is a 4-dimensional matrix consisting of space coordinates x, y, z in millimeters and time point units of 2 seconds. The first 3 dimensions of the matrix code a volumetric image. The coordinates of a given matrix cell provide the voxel's location, and the value of the cell is the BOLD signal strength. The 4th dimension accounts for the value at each voxel in time.

Standardization of fMRI Data

Before proceeding, we must change the fMRI data matrix from 4D to 2D. The NiftiMasker from the NiLearn module changes the shape of the data to n samples x n features.

The goal is to unfold the first 3 dimensions that code for location into 1 dimension. We are retaining the BOLD signal at a particular voxel, while discarding the information about the voxel's location. We end up with a long list of values for each voxel per time point. We also retain time information. This results in a 2D matrix.

In the example file, the shape changes from (27307, 1, 6, 180) to (180, 163842). The 180 time points were retained. The voxels were collapsed by multiply the first three columns (27307 x 1 x 6 = 163842 voxels).

Cleaning up Data and Assigning Labels

After standardizing, information stored in paradigm files, which contains the stimuli schedule, is used to discard activity we are not interested in (such as instructions and fixation periods) and assign labels to activity of interest. The goal is to store all the information in a master matrix, which we can later subset to perform various analyses and test multiple hypotheses.

Matrix structure:

Columns:	Participant	Stimulus Type	Task Type	BOLD signal
Values:	Participant ID [0-7]	Dog/ Cat sound Male/ Female face [1-4]	Active Passive [1-2]	Large NumPy array Float type

III. Basic Statistics

The dataset is presently too large to process as a single matrix on my personal computer. I will attempt to reduce features or access a more powerful computer to perform the standardization of data and its transfer to a master matrix. Thus, the statistics computed for this assignment are based on all the run files for the first subject. This consists of 16 files from 8 runs corresponding to left and right hemispheres of the brain. Each file contributes 8 rows of brain activity throughout a 40-second span of time (20 consecutive time points), in which each participant was exposed to a stimulus.

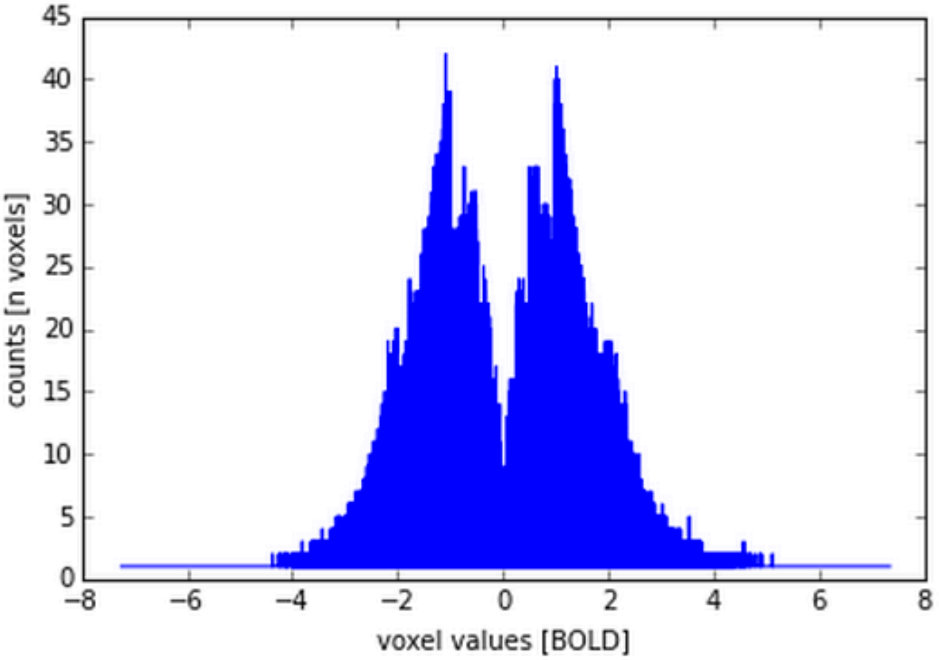
The statistics were computed only on the values corresponding to the BOLD signal because the first three columns are labels and identifiers.

Density of Values

Rows were filled in to force the rows of the master matrix to match. The columns of the matrix are as big as the largest file, so the smaller files have 0 values placed in after the real voxel values ended. Since 0 values are likely to be null values, they were replaced with NA. I will attempt to run SVD on the matrix with NA values in order to reduce dimensionality. This will hopefully be enough to reduce the influence of these missing values in the classification.

Distribution of Values

```
[<matplotlib.lines.Line2D at 0x109ad2550>]
```



Miscellaneous Statistics

Mean: -0.00174036393241

Median: 0.00417875405401

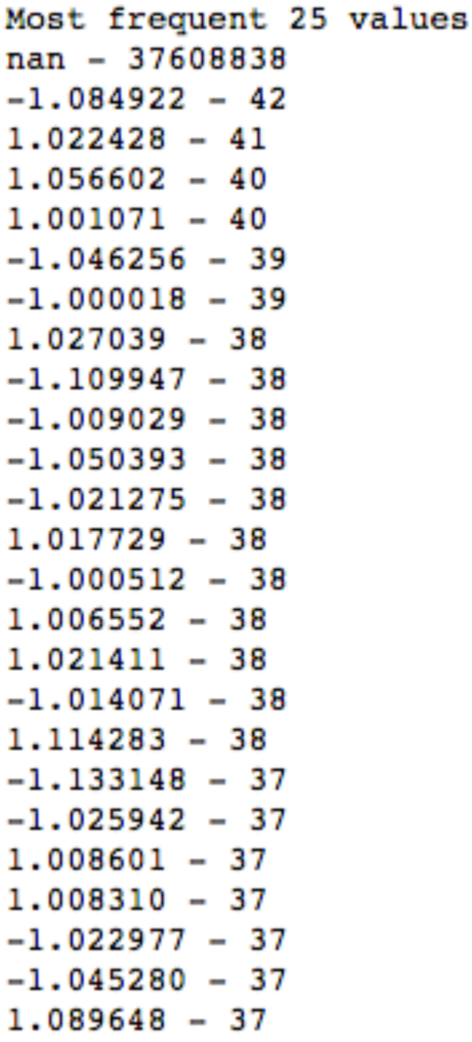
Standard Deviation: 0.97441313323

Variance: 0.949480954211

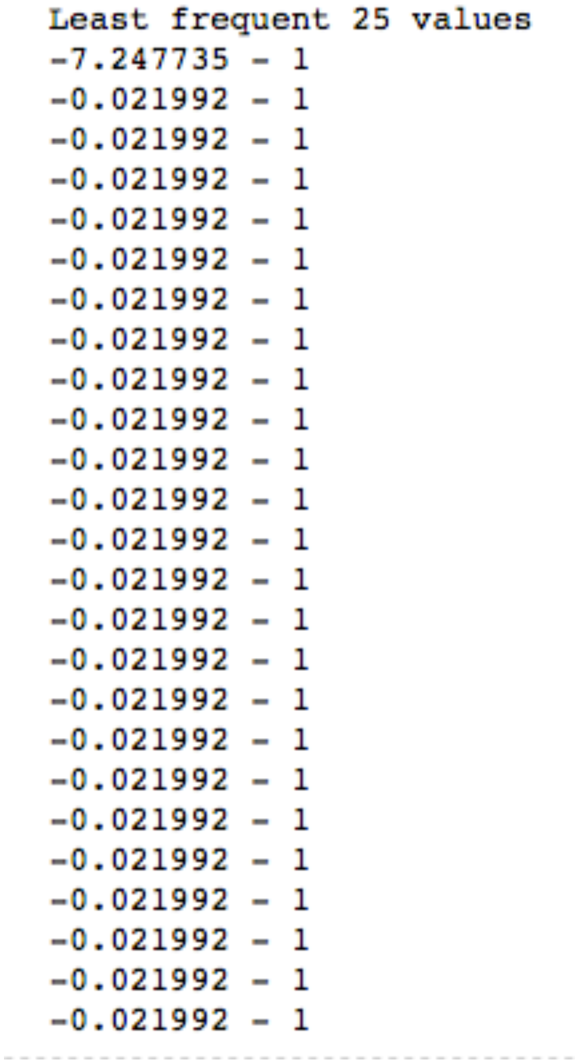
Although the mean and median point to a distribution centered on 0, which would be troublesome since all 0 values were converted to NA, the histogram shows this is a bimodal distribution. Values decrease as they approach zero, and the most frequent values are close to ±1.

By observing the most and least frequent values, we can see that the ‘nan’, which were once 0 values, are disproportionally frequent in comparison to other values.

Most and Least Frequent Values



```
Most frequent 25 values
nan - 37608838
-1.084922 - 42
1.022428 - 41
1.056602 - 40
1.001071 - 40
-1.046256 - 39
-1.000018 - 39
1.027039 - 38
-1.109947 - 38
-1.009029 - 38
-1.050393 - 38
-1.021275 - 38
1.017729 - 38
-1.000512 - 38
1.006552 - 38
1.021411 - 38
-1.014071 - 38
1.114283 - 38
-1.133148 - 37
-1.025942 - 37
1.008601 - 37
1.008310 - 37
-1.022977 - 37
-1.045280 - 37
1.089648 - 37
```



```
Least frequent 25 values
-7.247735 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
-0.021992 - 1
```

Hypotheses

Decoding consists of using brain image to predict external variables, such as an individual's thoughts. I will train a classifier to decode BOLD signal values and predict the modality of stimulus (visual or auditory), stimulus (cat vs. dog sounds or male vs. female faces), and what task (active vs. passive) the participant was performing.

H1: A classifier can predict above chance (50%) whether participants were shown a sequence of visual or auditory stimuli across subjects. The classifier will be trained on data from various participants and will be able to predict for various participants.

H2: A classifier can predict above chance (50%) whether participants were engaged in an active, working memory task or whether they were passively shown stimuli across subjects. The classifier will be trained on data from various participants and will be able to predict for various participants.

H3: A classifier can predict above chance (50%) whether participants heard cat or dog sounds for an individual. The classifier will be trained on data from a particular individual and will be able to predict for that individual.

H4: A classifier will not predict above chance (50%) whether participants heard cat or dog sounds across subjects. A classifier trained on data from various participants attempting to predict for various participants will fail because object concept representations in the brain vary widely across individuals.

H5: A classifier can predict above chance (50%) whether participants saw a male or female face image for an individual. This indicates that the classifier will be trained on data from a particular individual and will be able to predict for that individual.

H6: A classifier will not predict above chance (50%) whether participants saw a male or female face image across subjects. A classifier trained on data from various participants attempting to predict for various participants will fail because object concept representations in the brain vary widely across individuals.