Comparing Healthy and Tumor Mouse Resting State Functional Networking: A Machine Learning Approach

Darian Hadjiabadi
The Johns Hopkins University
Department of Biomedical Engineering
dhadjia1@gmail.com

Jacob Pines
The Johns Hopkins University
Department of Computer Science
jpines2@jhu.edu

Abstract

Functional MRI (fMRI) is a well proven procedure for acquiring the Blood Oxygen Level Dependent (BOLD) signal resulting from neural activation and subsequent hemodynamic response. Correlating this signal between brain regions gives insight on the *functional connectivity* between them. In this work, a supervised neural network and unsupervised k-means++ clustering algorithms were implemented for healthy and tumor mouse models. Both techniques are a means for understanding differences in functional architecture between healthy and tumor subjects. Classification between healthy and tumor voxels achieved accuracy > 0.9 for all tumor datasets, indicating effective classification even for small tumors. Results from k-means clustering over 10 clusters show significant differences in functional organization between healthy and tumor subjects, however it may be the case the results are effected by the initial clustering conditions. Regardless, evidence does not suggest significant changes over varying tumor mass.

1 Introduction

Temporal synchrony of the BOLD signal between various brain regions has been the foundation for the field of functional connectivity. The BOLD signal is a combination of endogenous brain fluctuations; including but not limited to vasomotor, electro physiological, cytochrome, and random noise(Kiviniemi, 2008). Changes in the BOLD signal has been correlated to neural activity, which, through the still poorly understood neurovascular coupling mechanism, is responsible for local increases in blood flow and blood volume for the purposes of oxygen consumption(Hillman, 2014)(Iadecola and Nedergaard, 2007). Ogawa et. al showed in his seminal paper that T_2^* MRI is sensitive enough to capture BOLD signal fluctuations, which varies by only 2% during neural activation(Ogawa, 1990). This procedure, known as fMRI, is the basis for understanding the vast interactions between brain regions.

The field further expanded when Biswal et.al showed highly correlated low frequency fluctuations (0.01 - 0.1 Hz) in motor cortex within resting state subjects(Biswal, 1995). Despite not receiving initial attention, further studies were able to reproduce Biswal's results nearly half a decade later. This branch of functional connectivity has since been termed resting state functional connectivity (rs-fc). Effort has been made to map default mode circuits, and to observe how these circuits vary between healthy and diseased subjects(White, 2011)(Bartolomei, 2006)(Guilfoyle, 2013).

Understanding functionally connected circuits resulting from resting state brain activity has come in the form of well studied machine learning tools. PCA and ICA take a data driven approach to find uncorrelated (PCA) and independent (ICA) sources of variance(White, 2011)(Holschneider, 2014)(Jonckers, 2011). Both have proven useful in finding a number of circuits without any prior knowledge. Neural networks have been

implemented to observe differences in underlying functional structure between healthy, bipolar, and schizophrenic subjects(Mamah, 2013). Furthermore, k-means clustering has also been utilized as a way to explore differences in BOLD signal under tumor conditions(Baudelet, 2003). The underlying principle of these works have been to understand how functional connectivity is effected under various diseased conditions.

This paper aims to dabble in the likes of topics such as these, using machine learning and functional connectivity as a means for observing changes in global functional structure between healthy mice and tumored mice subjects. To do so, a supervised 2-layer neural network is implemented with the goal of discovering a well defined decision boundary that separates tumor from healthy functional state space. Furthermore, unsupervised k-means clustering is implemented to analyze how cluster sizes, and hence spatial organization, differ between healthy and tumor subjects.

2 ML Techniques

Data

7 Healthy and 7 tumored mouse subjects were subject to T_1 weighted anatomical MRI in addition to T_2^* weighted functional MRI while anesthetized. In addition, region of interest (ROI) label fields were provided for each subject, divided into hippocampus, neocortex, olfactory, brainstem, thalamus, striatum, and hypothalamus. Under the label fields, each voxel's BOLD time course was correlated with the mean BOLD signal of an ROI. This produces correlation value activation maps. Each ROI produces a unique activation map, signifying the functional connection between that ROI and the brain voxels. The input for the machine learning programs is a voxel's correlation values to each ROI's mean time course. Below is an example.

```
1:  -.018566 2:  0.227906
3:  -0.027379 4:  0.026997
5:  -0.036589 6:  0.035579
7:  -0.152997 8:  0.047457 9:
-0.71873 10:  -0.134095 11:
-0.079051 12:  -0.049998 13:
-0.128393.
```

Neural Network

A neural network is a type of universal function approximation which is constructed as a series of weighted interconnected "neurons" passing information between each other(Bishop, 2007). This useful feature will ideally be able to find decision boundaries between healthy and tumor functional state space. One of the ways to classify data which is non-linear is by learning additional features. Neural networks work by taking a dataset, weighting the data, and passing the data through a non-linear function. Given a data set $x = \{x_i\}_{i=1}^N$ where $x_i \in \mathbb{R}$, we can create non-linear features in x by doing the following:

- Start with a linear function of x, $w \cdot x$.
- Pass the output of the linear function through a non-linear function:

  $$z = h(w \cdot x)$$

  where h is a non-linear function like the logistic function.

z is the output of a non-linear function, combinations of z are non-linear in x. Each of these mappings from (w, x) to z represents a perceptron, which is a linear classifier that can be chained together with other perceptrons to form non-linear classifiers.

Training

The method we use for training is the backpropogation algorithm. The method has two main steps. First, we use a forward pass to compute the output bottom-up. Then, we use a backward pass that computes the derivatives of all intermediate quantities top-down.

Our neural network is represented by the function $\hat{y} = f_\theta(x)$, where x is the input and θ is the weights and neon biases of the network. The loss function is $J = \frac{1}{2}(\hat{y} - y^*)^2$. Training of the neural network is equivalent to finding parameters θ that minimize the objective function over the entire training set $\{x_i, y_i\}_{i=1}^N$:

$$\theta^* = argmin \Sigma_{i=1}^N E_i(\theta)$$

where

$$E_i^\theta(\theta) = l(f_\theta(x_i), y_i^*)$$

In order to minimize this objective iteratively using out training data, we use stochastic gradient descent [Artificial Neural Network Notes]:

1. Choose a starting point θ.
2. While not converged:
   - Choose a step size $\eta_t > 0$.
   - Sample a training instance i.
   - Take a small step following the gradient down.

$$\theta^{(t+1)} = \theta^{(t)} - \eta_t \delta E_i(\theta)$$

or

$$\theta_j^{(t+1)} = \theta_j^{(t)} - \eta_t \frac{dJ}{D\theta_j}$$

Prediction

Given an instance x, predict the output $\hat{y} = f_\theta(x)$. For out model with 1 output and D nodes in the hidden layer, we predict the output:

$$\hat{y} = h^{(2)}(\Sigma_{j=1}^{D} w_j^{(2)}(\Sigma_{i=1}^{M} w_{ji}^{(1)} x_i + w_{j0}^{(1)}) + w_0^{(2)})$$

K-means Clustering

Unsupervised clustering is a useful method for uncovering the structure of unlabeled data. In context of the problem, unsupervised clustering is an automated tool for determining how the global functional connectivity architecture differs between healthy and tumored subjects. Each cluster contains examples exhibiting a similar trend in ROI BOLD signal correlation values. Observing changes in cluster sizes gives insight on how the correlation based state space is reorganized under a tumor mass. Initial clusters were chosen via k-means++ algorithm, which uses actual data points as cluster centers. This seeding method yields considerable improvement in the final error of k-means. Furthermore, convergence occurs much more quickly despite initial computational investment(Arthur, 2007). The algorithm for picking initial clusters is briefly given below.

1. Choose one center uniformly at random
2. For each data point x, compute $D(x)$, the distance between x and the nearest center
3. Choose new data point at random to be the new center, where each data point has $p(x) \propto D(x)^2$.
4. Repeat steps 2,3 until k centers have been chosen
5. Proceed with standard k-means clustering

For ideal separation of state space, it is necessary to pick initial clusters that are as far in distance as possible. With this in mind, the probability of data point x_i being chosen is set to $p(x_i) = \frac{1-e^{-[\alpha * D(x_i)^2]}}{Z}$ where Z is normalizing factor equal to $\sum_{j=1}^{N}(1 - e^{-[\alpha * D(x_j)^2]})$. Figure 1 shows the $p(x) * Z$ over varying values of α. Observe that smaller distances are given significantly smaller weight when compared to larger distances. An ideal α value needs to consider the size of the state space as well as the number of dimensions. The k-means algorithm on each

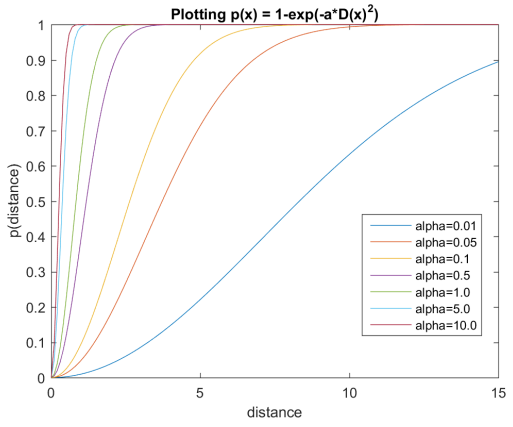

Figure 1: Changes in $p(x) * Z$ over α

iteration updates clusters by finding points that are maximally similar. Define $r_{nk} = 1$ if x_n is in cluster k, otherwise 0. Define the mean of cluster k as $\mu_k = \frac{\sum_{i=1}^{N} r_{nk} x_n}{\sum_{i=1}^{N} r_{nk}}$. In order to maximize the similarity of every cluster, choose a cost function $J(r, \mu) = \sum_{n=1}^{N}\sum_{k=1}^{K} r_{nk}||x_n - \mu_k||^2$. To solve for r_{nk}, let $\frac{\partial J}{\partial r_{nk}} = 0$, to obtain $r_{nk}^* = 1$ if $k = argmin_j||x_n - \mu_j||^2$, else 0. To solve for μ_k, let $\frac{\partial J}{\partial \mu_k} = 0$, to obtain $\mu_k^* = \frac{\sum_{i=1}^{N} r_{nk}^* x_n}{\sum_{i=1}^{N} r_{nk}^*}$. It should be noted that J is a non-convex function causing the clustering algorithm to be an NP-hard problem. Final results will be determined by initial conditions. Convergence is achieved when each updated cluster is less than distance ϵ from its

previous location. That is, if *any* cluster k has distance $||\mu_k - \mu_k^*||^2 > \epsilon$, the system has not converged.

3 Work Done

Initial work was done to process the raw tumor and healthy data. *A*nalysis of *F*unctional *N*euro*I*mages (AFNI), an open source software developed by the University of Wisconsin, was used at this phase. AFNI provided a quick and easy way to feed in raw fMRI data, calculate mean time courses, produce correlation maps, and dump out the resulting correlation values. A python script was used to concatenate the correlation maps into data to be fed into both the neural network and k-means clustering algorithms. The exact nature of the data and correlation vector is found in the *Data* subsection of *ML Techniques*.

Both the neural network and k-means clustering algorithm were coded from scratch in Java. One challenge with implementing the neural network was ensuring that all of the matrices were setup properly so that when doing matrix multiplication and addition, the correct weights, biases, and gradients were being multiplied and added. We started off having the second of three dimensions in the weights matrix represent the lower nodes in the network and the third dimension the upper nodes they were attached to. We realized that the opposite representation made more sense for how matrices are multiplied and the gradient is calculated. Our second issue was a small coding error where the sigmoid function was calculated incorrectly, which led to incorrect results and hours of debugging. The last issue was our choice of loss function. Initially we were using $\hat{y} - y^*$, but switched to $\frac{1}{2}(\hat{y} - y^*)^2$ and realized much better results. K-means clustering was initially implemented with initial clusters being spread across the diagonal. However, due to the curse of dimensionality, only two clusters would ever contain data points. Therefore it was opted to use an initial clustering method (k-means++ algorithm) that suggested a probabilistic method of generating cluster centers by taking into account the spread of the data rather then the spread of the entire space. Another problem with the k-mean clustering, implemented as is, is the superficial understanding of how the state space is reorganized under diseased conditions. Clearly more robust clustering algorithm needs to be implemented to not observe a change, but to also observe where these changes are occurring.

Classification accuracy tests were run on the neural network. In addition, several layers of experiments were run on k-means clustering. This included determining ideal cluster size, α parameter, culminating with tests on tumor datasets. *tcsh* testing scripts can be found in the src/ directory of the submitted archive. K-means clustering produced large amounts of data that had to organized and interpreted via python scripts. Student t-tests were run on tumor clustering output to test for statistical significance between tumor and healthy datasets. Our submitted archive file is only a portion of the full extent of the work. If you wish to view the complete work, which includes raw data, output data, processing scripts, and excel files, please check `www.github.com/jpines2/MachineLearningFinal`

4 Results

As reference, the table below shows range of tumor sizes in increasing order.

dataset	volume (mm^3)
m30330	2.67
m20329	10.54
m50329	21.20
m10330	21.42
m40329	40.04
m40330	96.12
m20330	121.22

Classification

Testing was done on a 2-layer neural network to analyze whether or not a correlation vector could be determined as having come from a tumor dataset or a healthy dataset. Overfitting was reduced by utilizing 7-fold cross validation, implemented on a randomly mixed healthy and tumor dataset. Figure 2 shows classification accuracy over increasing tumor size. 20 experiments were run for each dataset to determine mean accuracy and variance. Learning

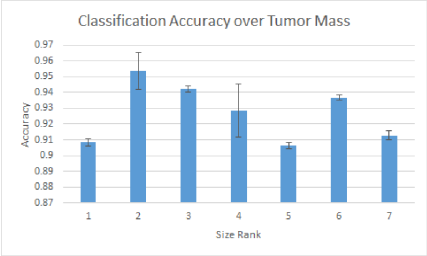

Figure 2: Classification Accuracy over Tumor Mass Size

parameter was set to 0.01. In all instances, mean accuracy of > 0.9 was observed and standard deviation did not exceed 2%. This tight standard deviation can be attributed to the sheer number of data that the neural net had to train on, resulting in good model generalization. In addition, this shows that having a single hidden layer of abstraction was sufficient for classification. More layers can be added, though may not necessarily produce significantly better results. While excellent classification was obtained across all datasets, no obvious trend could be found between tumor size. All attempts to fit a trend to the mean across tumor size produced R^2 close to 0. However, there is a positive note to this. It was originally hypothesized that it might be easier to classify voxels from a large tumor model. Observing high classification across tumors of varying sizes shows the neural network is sensitive to even extremely small tumors, which may be useful for early detection.

Unsupervised Clustering

5-cluster means was initially done on healthy data to observe healthy cluster state-space distribution. Below is an example of what is output by the program.

```
cluster 1:  0.197212
cluster 2:  0.216076
cluster 3:  0.213032
cluster 4:  0.159421
cluster 5:  0.214189
```

100 tests were done, each test outputting the percent of voxels in each cluster. Mean cluster sizes across were obtained along with the variance. Figure 3 shows the uniform distribution of voxels across clusters. Initially it was suspected that

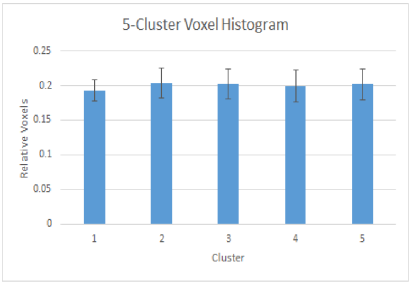

Figure 3: 5-Cluster Histogram Distribution

the distribution would be Boltzmann, though the idea was flawed as it assumed that the algorithm somehow knew whether or not certain voxels were intrinsically different. This is clearly not the case as the algorithm makes no comparison of such sort - rather it simply compares the euclidean distance from cluster center to point. If you imagine capturing a packet of pure white light and binning the photons based on wavelength, you would acquire a nearly uniform distribution. Likewise, we are capturing healthy voxels in their most natural fluctuating conditions - the uniform distribution makes sense. In order to determine the ideal cluster size for testing on tumor datasets, the same 100 tests were ran on k-cluster algorithm for k = $[5, 10]$. The mean sum of squared error between the output histogram and an ideal histogram was acquired and compared, shown in Figure 4. Tests across $k = [5, 10]$ were

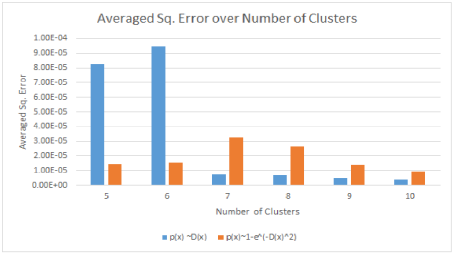

Figure 4: average sum of sq error across number of clusters

initially done with initial cluster centers chosen with

probability $p(x_i) = \frac{D(x_i)^2}{\sum_{j=1}^{N}(D(x_j)^2)}$. In both cases, running the algorithm with 10 clusters produced minimal error (closest to the uniform distribution). However, the probability distribution, as mentioned in the ML techniques section, produced a much smaller variance, as seen in Figure 5. This yields

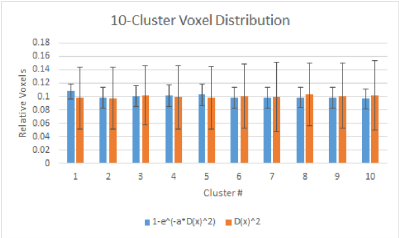

Figure 5: 10-cluster voxel distribution between initialization probability distributions

higher confidence each time a clustering distribution is sampled; it will be closer to the ideal uniform distribution when initializing with exponentially based probability distribution. Furthermore, testing was done on the α parameter to determine if α effected the variance of cluster distribution. 100 tests were done on $\alpha = 0.01, 0.05, 0.1, 0.5, 1.0, 2.0, 5.0, 10.0$ at 10 clusters. Figure 6 shows the normalized variance distribution, showing very little difference between the α parameters. Thus α was chosen to be 0.05 based on Figure 1 and careful consideration of the state space. 100 tests of 10-mean clustering

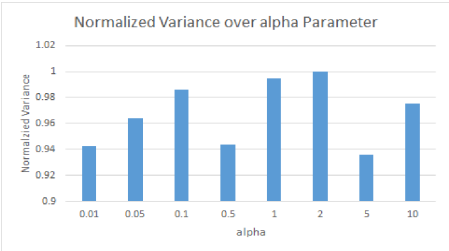

Figure 6: Cluster variance over α

were done on each of 7 tumor datasets with mean and variance obtained. On each test, the converged cluster centers from a healthy input was used as the converged cluster centers for the tumor datasets. Diseased data points closest to an input cluster center were assigned to that cluster. Figure 7 shows the distribution over clusters, with intra-cluster columns assigned in increasing tumor size. Welch's

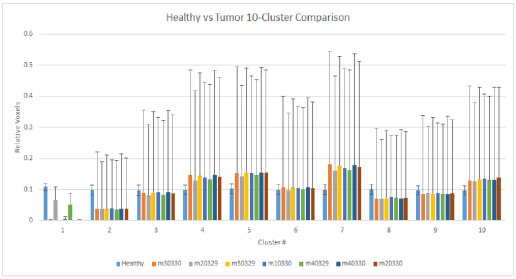

Figure 7: Percent voxels in each cluster over tumor datasets

two-sample t-test showed extremely statistically significant cluster distribution ($p < 0.001$) for clusters 1, 2, and significantly significant ($p < 0.05$) cluster distribution for cluster 7, across all tumor datasets when compared to the healthy results. In all cases, the deviation from the uniform distribution is clearly observed. In addition, standard errors were orders larger then its healthy counterparts, bringing into question whether or not probabilistically determined initial clusters were ideal. However, this figure also eludes to the possibility that there is a difference in state space distribution between healthy and tumored subjects, though there appears to be no observable difference in distributions between tumor masses of varying sizes. Further testing will need to be conducted to prove such claims.

5 Conclusion and Future Work

Classification gave excellent results across all tumor datasets. This indicates a clear decision boundary was found between healthy and tumor state space. While this initially did not fit what was expected, sensitivity to small tumors may have its own benefits.

Clustering results have shown uniform distri-

butions for healthy datasets and non-uniform distributions for tumor datasets. This gives insight into how functional connectivity deviates from normal under tumor conditions, but no evidence is presented on where these changes occur. In addition, no evidence suggests differences in spatial organization amongst tumors of varying sizes.

Future work in this area will deterministic initial condition methods. In addition, a Gaussian mixture model may be implemented to remove hard assignments, allowing for more generalization. Furthermore, observing where functional changes are occurring will be necessary for more in depth analysis.

Comparison to Proposal

- Opted to use a two layer neural network rather then a deep neural net. Implementation was successful.
- Trained and tested the neural network on healthy and tumor data, acquiring good classification accuracy.
- Implemented a k-means clustering algorithm. Not in our original proposal but provided very useful.
- Did not use alternative parameter spaces.
- Overall we surpassed our original expectations.

Disclosure

All data and results belongs to *The Pathak Lab*. Use of data for outside work is not permitted.

References

Ogawa S. 1990. *Brain Magnetic Resonance Imaging with Contrast Dependent on Blood Oxygenation.* Proc. Natl. Acad. Sci, 89:9868–9872

Biswal Bharat, Hyde S. James. 1995. *Functional Connectivity in the Motor Cortex of Resting Human Brain Using Echo-Planar MRI.* Magnetic Resonance in Medicine, 34:537–541

Kiviniemi Vesa. 2008. *Endogenous Brain Fluctuations and Diagnostic Imaing.* Human Brain Mapping, 29:810–817

Hillman M.C. Elizabeth. 2014. *Coupling Mechanism and Significance of the BOLD Signal: A Status Report.* Annual Review of Neuroscience, 37:161–181

Iadecola Costantino and Nedergaard Maiken. 2007. *Glial regulation of the cereberal microvasculature.* Nature Neuroscience,10(11):1369–1376

Guilfoyle N. David. 2013. *Functional Connectivity fMRI in Mouse Brain at 7T Using Isofluorane.* J. Neuorscience Methods,214(2):144–148

Bartolomei Fabrice. 2006. *Disturbed Functional Connectivity in Brain Tumors Patients: Evaluation by Graph Analysis of Synchronizaton Matries.* Clinincal Neurophysiology, 117:2039–2049

Holschneider P. Daniel. 2014. *Functional Connectivity-based parcellation and connectome of cortical midline structures in the mouse: a perfusion autoradiography study.* Frontiers in Neuroscience,61(8):1–18

White R. Brian. 2011. *Imaging of Functional Connectivity in the Mouse Brain.* PLoS ONE,6(1):1–10

Jonckers Elisabeth. 2011. *Functional Connectivity fMRI of the Rodent Brain: Comparison of Functional Connectivity Networks in Rat and Mouse.* PLoS ONE

Mamah D. 2013. *Resting state functional connectivity of five neural networks in bipolar disorder and schizophrenia.* J. Affect Disorders,150(2):601–609

Baudelet C. 2003. *Cluster analysis of BOLD fMRI time series in tumors to study the heterogeneity of hemodynamic response to treatment.* Magnetic Resonance in Medicine,49(6):985–90

Arthur D. 2007. *k-means++: the advantages of careful seeding.* Society for Industrial and Applied Mathematics,1027–1035

Bishop Christopher 2007. *Pattern Recognition and Machine Learning.* Chapter 5, Neural Networks