# Machine Learning and Feature Selection Analysis on Dynamic Connectivity Signatures in Neural Networks Engaged for Emotion Regulation

Emily Paul

October 1, 2019

# Contents

# Chapter 1

# Introduction

Major Depressive Disorder (MDD) is a mental disorder characterized by persistent low mood, loss of interest, and a reduction in the ability to regulate emotion. Its prevalence, especially in younger demographics, makes it a growing concern [1]. Unfortunately, many of the criteria for diagnosis listed in the Diagnostic and Statistical Manual of Mental Disorders, Fourth Edition, such as feelings of guilt, difficulty concentrating, and suicidal ideation, are highly subjective and difficult to quantify. This lack of conclusive biomarkers makes this disorder difficult to diagnose consistently [5]. This study aims to discover some quantitative features that can aid in the classification of potential patients. Creating a machine learning model that can quantitatively and reliably distinguish between patient and non-patient data would greatly improve the diagnosis process. Over the course of this project, some preliminary machine learning and feature selection analysis was conducted on a range of subject data including functional Magnetic Resonance Imaging (fMRI) time series encapsulating changing levels of activity in different brain regions during emotional regulation.

All scripts referred to in the report can be found in the MachineLearning_NeuroscienceProject repository at https://github.com/VeraLiconaResearchGroup/MachineLearning_NeuroscienceProject.git.

# Chapter 2

# Background Information

## 2.1 Neuroscience Data Collection

Dr. Mike Stevens and his team at the Olin Neuropsychiatry Research Center collected data from 129 adolescents ranging in age from 12 to 18 years. 105 of the subjects had no history of mental disorders and were classified as healthy. The remaining 24 subjects were in remission from MDD. For each subject, the team recorded phenotypic data by administering tests such as the Beck Depression Inventory, the Early Adolescent Temperament Questionnaire, and the Multidimensional Anxiety Scale for Children; the full list is included in the MachineLearning_NeuroscienceProject repository. The researchers also collected data from 4 Functional Magnetic Resonance Imaging (fMRI) scans of the subject's brain. Where MRIs have been used to visualize the anatomical structure of the brain, fMRIs measure changes in cerebral bloodflow, and thus produced data in time series encapsulating the changing levels of metabolic activity in different brain regions. Over the course of each scanning period, the subject was shown a series of images, each designed to evoke an emotional response. During two of the scans, subjects were asked to heighten their emotional response to the presented stimuli, and during the other two scans, they were asked to repress the response to the best of their ability. Thus, the team produced 4 time series per subject, two each for increased and decreased emotional response.

## 2.2 Data Processing

Dr. Paola Vera-Licona and her team at the Center for Quantitative Medicine at UCONN Health created a pipeline to extract a dynamic conectivity signature (dynamic signature) from each time series of fMRI data. They represented each time series as a probabilistic boolean network (PBN): a collection of the possible boolean networks that could solve for the fMRI time series. They then represented each PBN as an adjacency matrix, $[a_{ij}]$,
where $a_{ij} = \dfrac{\text{number of times node i is input to node j}}{\text{total number of inputs to node j}}$.
As shown in Fig. 2.2.1, these matrices can be visualized as graphs in which nodes are brain regions and edges are interactions between them. An edge exists between node $i$ and node $j$ if $a_{ij} \neq 0$. In the graph in Fig. 2.2.1, both node size and node color are mapped to betweenness centrality, which is a measure of the degree to which a node falls in the shortest path between any two other nodes. Large, purple nodes have high betweenness centrality, and thus act as hubs in the network, while small, yellow nodes have low betweenness centrality. Similarly, edge density is mapped to edge betweenness; heavier edges have higher edge betweenness. The networks were visualized this way to make them easier to interpret.

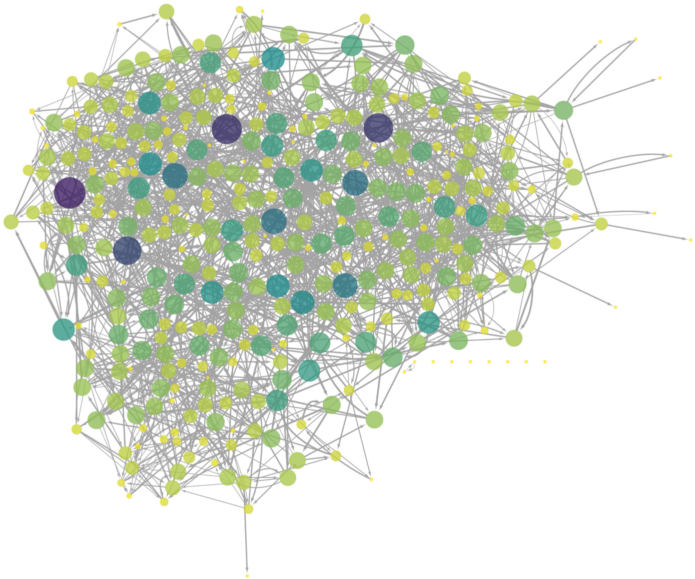


Figure 2.2.1: Dynamic signature graph

# Chapter 3

# Data Mining

## 3.1 Conserved Components in Dynamic Signatures

After checking for intersections among the dynamic signatures (see the script creating_intersection_networks.R), it was observed that there were several subgraphs which were conserved across over 75% of the dynamic signatures. Of these, the connected components shown in Fig. 3.1.1 were of particular interest. They involve the same brain regions and display identical connectivity, in the right and left hemispheres. These components illustrate activity in the amygdala-hippocampal complex, which has been established to be a driver of emotion regulation. It controls the formation of episodic memories in response to stimuli and modulates cognitive appraisal [7]. Studies have uncovered some potential associations between amygdala [3] and hippocampus [2] core volume and MDD.

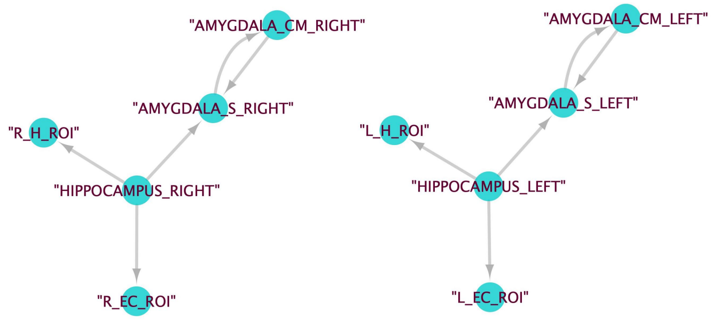


Figure 3.1.1: Conserved amygdala-hippocampal complex

## 3.2 Features Generated for Dynamic Signatures

### 3.2.1 Network Features

Network features were gathered using the iGraph package in R. All of the datasets used in the analysis can be constructed by running the scripts in the MachineLearning_NeuroscienceProject repository, starting with the construction of iGraph objects in the script creating_iGraph_objects.R.

#### General Network Characteristics

16 network characteristics were calculated for each dynamic signature. The list of characteristics can be found in the script general_network_features.R. The INCgraphstats dataset contains the characteristics calculated for both increased emotional response dynamic signatures per subject, and thereby has 32 features. The DECgraphstats dataset contains the characteristics calculated for both decreased emotional response dynamic signatures per subject.

#### Motif Occurrences

Two subgraphs were constructed for each dynamic signature based on the location of the hippocampus region (left or right); each subgraph consisted of one of the hippocampus regions and all of the nodes of first and second order connectivity to it. For each subgraph, the number of occurrences for each of 16 different motifs (each composed of 3 vertices) was calculated; see the script triad_occurrences_hippocampus_subgraphs.R.

### 3.2.2 Phenotypic Data

Dr. Stevens and his team produced a set of 531 phenotypic characteristics per subject. Unfortunately, only 13 of those were complete; the remainder contained missing values for various subjects. Instead of introducing bias and potentially skewing results by introducing false data points, only the 13 complete variables were used in the analysis. The list of the complete variables can be retrieved from the script clinical_data_processing.R.

### 3.2.3 Data Collation

The data was collated into a dataframe wherein each subject was a row and each feature was a column. All columns containing only zeros were removed before applying dimensionality reduction and machine learning techniques, leaving 203 features per subject; see the script subject_data_analysis.R. These features will subsequently be referred to as the subject data.

# Chapter 4

# Dimensionality Reduction and Machine Learning

## 4.1 Dimensionality Reduction

Principal component analysis (PCA) is a dimensionality reduction technique that produces a list of principal components which are linear combinations of the original variables. It is used to visualize high dimensional datasets in 2 or 3 dimensions by capturing as much variance as possible within the first 2 or 3 principal components [4]. The biplot for the PCA conducted on the subject data is shown in Fig. 4.1.1, and the scree plot is given in Fig. 4.1.2. They were computed using the prcomp() function in R on the scaled and centered subject data and visualized using Dr. Jason Cory Brunson's ordr package for the R tidyverse; see the script subject_data_analysis.R.

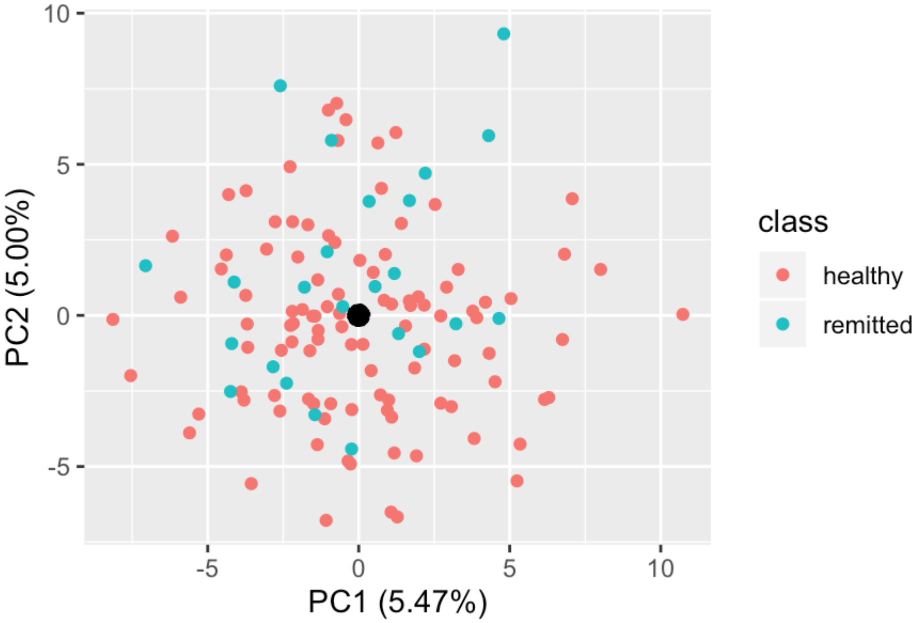


Figure 4.1.1: PCA biplot

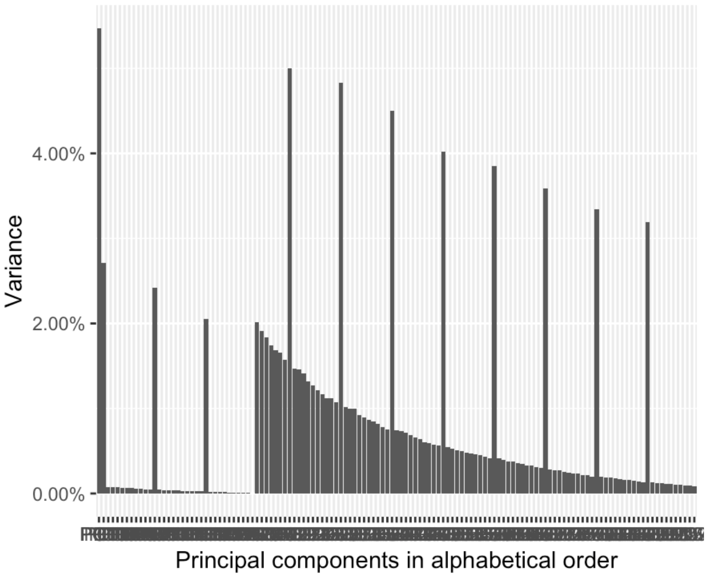


Figure 4.1.2: PCA scree plot (see 4.1.4 for component order)

The first principal component covered only 5.47% of the variance in the subject data, and the first 10 components cumulatively do not cover 50% of the variance. In addition, the first 2 principal components do not have significantly greater variance than the remainder of the components; this means the biplot visualization is not a good representation of the subject data. T-distributed stochastic neighbor embedding and multidimensional scaling were also conducted on the subject data, but were similarly unreliable.

PCA was also conducted on the INCgraphstats and DECgraphstats datasets. The biplots with labeled vectors and the scree plots are given below.

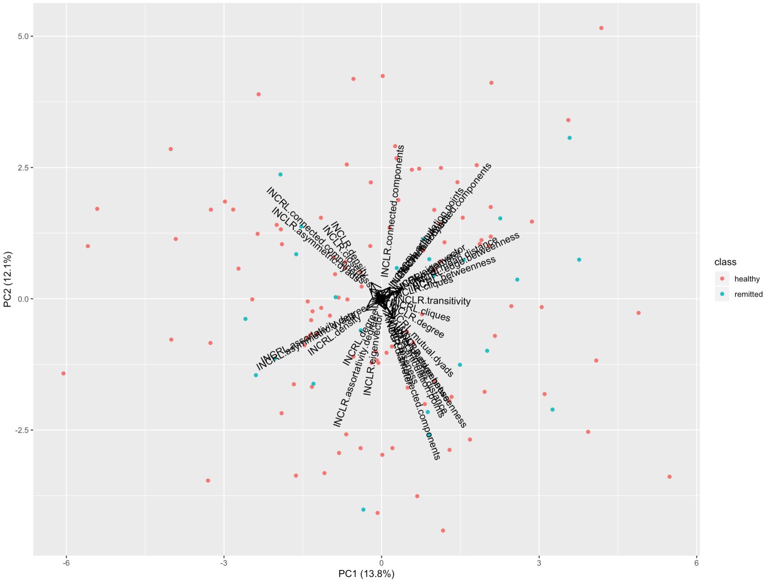


Figure 4.1.3: Increased emotional response general network characteristics PCA biplot

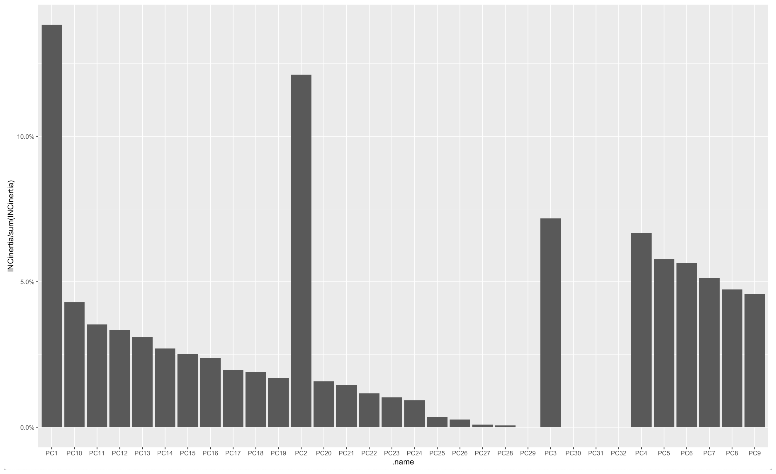


Figure 4.1.4: Increased emotional response general network characteristics PCA scree plot

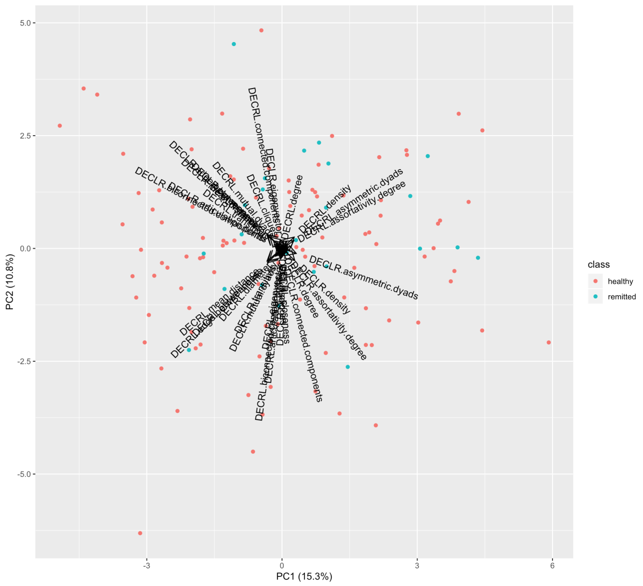


Figure 4.1.5: Decreased emotional response general network characteristics PCA biplot

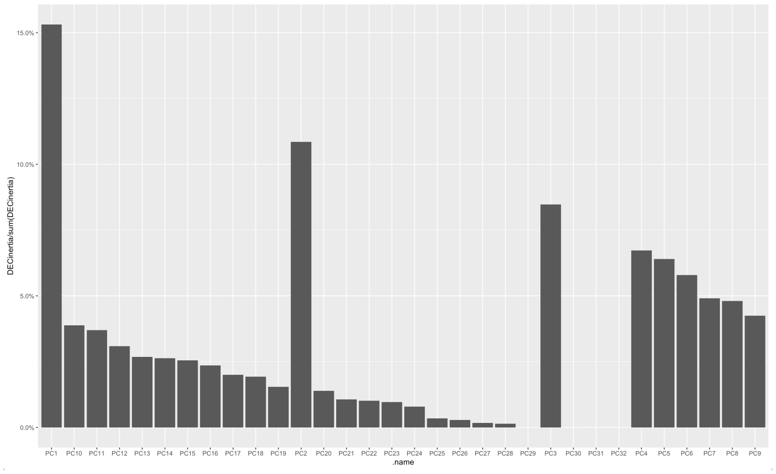


Figure 4.1.6: Decreased emotional response general network characteristics PCA scree plot

The first two principal components cumulatively captured over 20% of the variance in both the increased and decreased emotional response data, as shown in 4.1.4 and 4.1.6 respectively. This is still not enough to consider the biplots to be good representations of the data; conclusions should not be drawn from a biplot unless the first 2 components (if the plot is 2 dimensional) account for at least 50% of the variance in the data. At any rate, neither biplot (4.1.3 for increased emotional response and 4.1.5 for decreased emotional response) shows any clustering.

## 4.2 Machine Learning

### 4.2.1 Hierarchical clustering

Hierarchical clustering is an unsupervised machine learning technique where each point is initially assigned to its own cluster. The algorithm then iteratively clusters the two clusters who are the closest to eachother; this hierarchical clustering can be traced with a dendrogram [4]. The dengrogram for the hierarchical clustering conducted on the subject data using the hclust() function in R is given in 4.2.1.

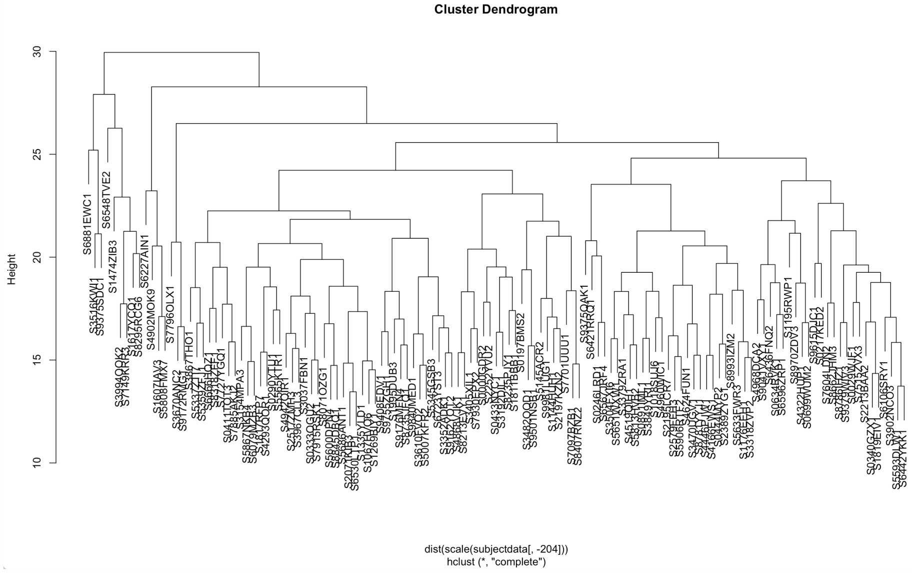
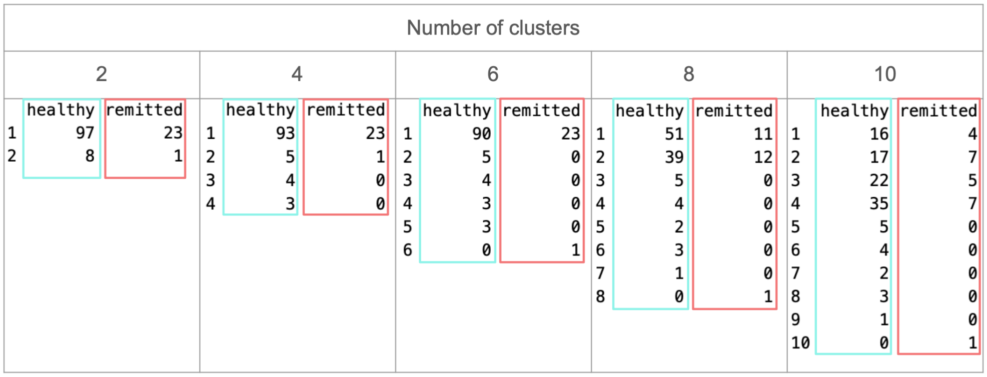


Figure 4.2.1: Dendrogram



| Number of clusters | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | | | 4 | | | 6 | | | 8 | | | 10 | | |
| | healthy | remitted | | healthy | remitted | | healthy | remitted | | healthy | remitted | | healthy | remitted |
| 1 | 97 | 23 | 1 | 93 | 23 | 1 | 90 | 23 | 1 | 51 | 11 | 1 | 16 | 4 |
| 2 | 8 | 1 | 2 | 5 | 1 | 2 | 5 | 0 | 2 | 39 | 12 | 2 | 17 | 7 |
| | | | 3 | 4 | 0 | 3 | 4 | 0 | 3 | 5 | 0 | 3 | 22 | 5 |
| | | | 4 | 3 | 0 | 4 | 3 | 0 | 4 | 4 | 0 | 4 | 35 | 7 |
| | | | | | | 5 | 3 | 0 | 5 | 2 | 0 | 5 | 5 | 0 |
| | | | | | | 6 | 0 | 1 | 6 | 3 | 0 | 6 | 4 | 0 |
| | | | | | | | | | 7 | 1 | 0 | 7 | 2 | 0 |
| | | | | | | | | | 8 | 0 | 1 | 8 | 3 | 0 |
| | | | | | | | | | | | | 9 | 1 | 0 |
| | | | | | | | | | | | | 10 | 0 | 1 |

Figure 4.2.2: Hierarchical clustering

The composition of each cluster can be visualized in a table; see the script subject_data_analysis.R. The results of various clusterings of the subject data is given in 4.2.2. No matter how many clusters the subject data is broken into, hierarchical clustering never produces a large cluster composed predominantly of one class or another. For example, consider cluster 1 in the columns for 2, 4, and 6 clusters; it contains 113 of the 129 subjects. The fact that the majority of the subjects here are healthy is merely a reflection of the makeup of the subject population. When this large cluster is split up (into clusters 1 and 2 in the column for 8 clusters), the remitted subjects are split evenly into the two resultant clusters. This indicates that the differences that the clustering is identifying do not divide healthy subjects from remitted. It may be useful to run a separate analysis on the subjects in certain clusters to determine what is causing them to be grouped together; the commands that produce lists of subject identifiers by cluster are in the script subject_data_analysis.R.

### 4.2.2 k-means

K-means is another unsupervised machine learning technique. The user specifies a number of clusters $k$ and the algorithm initially chooses k centroids. Initial clusters are determined by assigning each point to the nearest centroid, in of Euclidean distance. Each centroid is then shifted to the mean of its cluster, after which the points are reassigned to new clusters. The algorithm repeats this process until there is no change in the clusters between runs or until some previously defined stopping condition is met [4]. The kmeans() function was used in R.

Like with hierarchical clustering, the composition of each k-means cluster can be visualized in a table; see the script subject_data_analysis.R. The results of the k-means clustering display the same trends found in the hierarchical clustering: as $k$ increases, the remitted subjects are split evenly between the largest clusters.

### 4.2.3 k-nearest neighbors

K-nearest neighbors (KNN) is a supervised machine learning technique that classifies test cases according to their proximity to training cases. The user specifies the number of nearest neighbors $k$ to consider, and the algorithm classifies each test case as the class of the majority of the $k$ closest training cases [4].

Scripts were written to measure the accuracy of the KNN models by calculating their precision and recall,

where precision $= \dfrac{\text{true positives}}{\text{true positives} + \text{false positives}}$

and recall $= \dfrac{\text{true positives}}{\text{true positives} + \text{false negatives}}$.
These values were calculated for each $k$, after which precision-recall curves were plotted; see the scripts knn_binary_loop.R and precision_recall_curves.R. The precision-recall curve for the KNN models built from the subject data is given in 4.2.3.

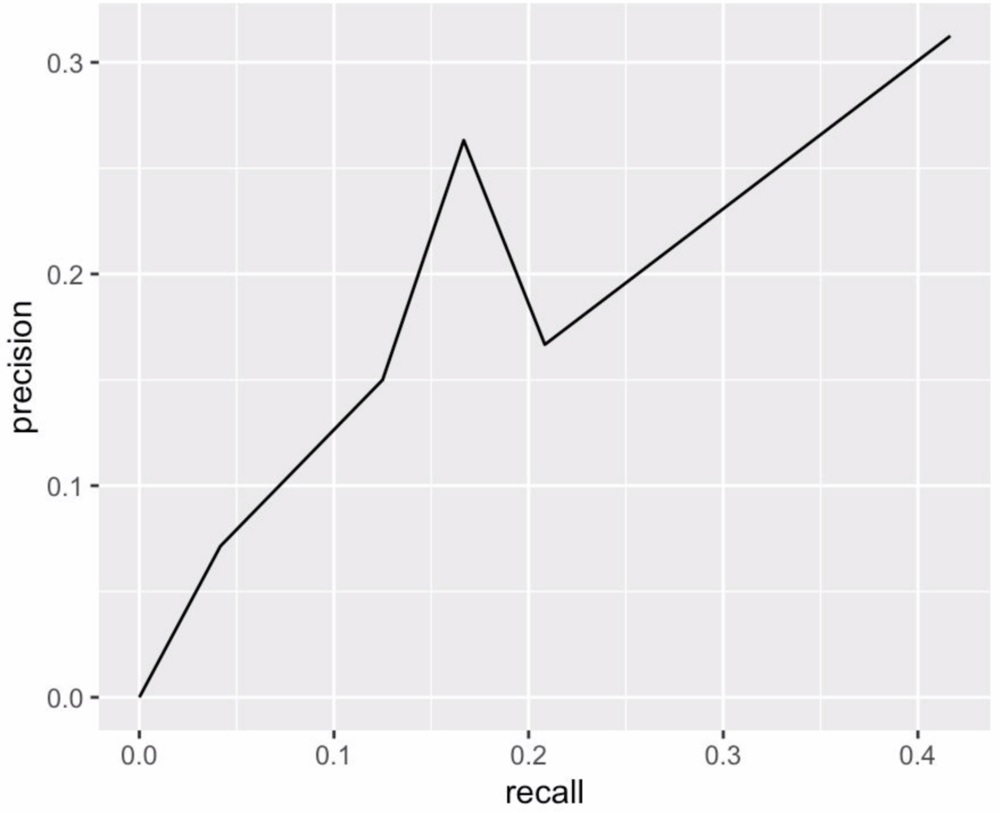


Figure 4.2.3: Precision-recall curve

A perfect model would predict with 100% precision and 100% recall. However, even the best KNN model only classified with approximately 30% precision and 40% recall; it is completely unreliable.

# Chapter 5

# Feature Selection

Machine learning techniques can be less effective on high dimensional datasets that have many irrelevant features. To attempt to reduce the noise in the subject data, feature selection was performed using the scikit-learn package in Python. A logistic model was used with recursive feature elminination, a wrapper function. It iteratively builds predictive models, ranks subsets of features based on model accuracy, and then removes the features with the lowest contributions from the existing set [6].

Three new datasets were constructed, one containing 80% of the total features, one containing 50%, and one containing 30%; see the script feature_selection.py. The analysis described in the Dimensionality Reduction and Machine Learning chapter was repeated on each of these datasets in R after running the script import_selected_features.R. For example, the biplot and scree plot for the PCA conducted on the 30% dataset are given in figures 5.0.1 and 5.0.2 respectively.

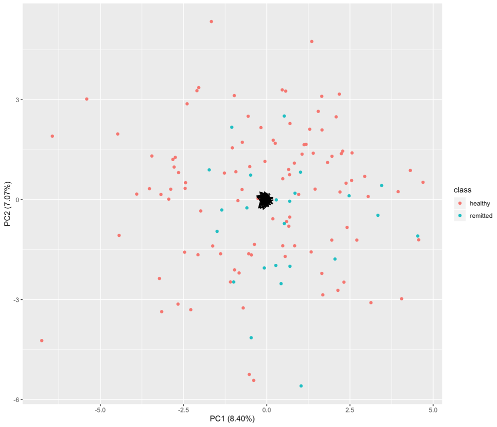


Figure 5.0.1: 30% dataset PCA biplot

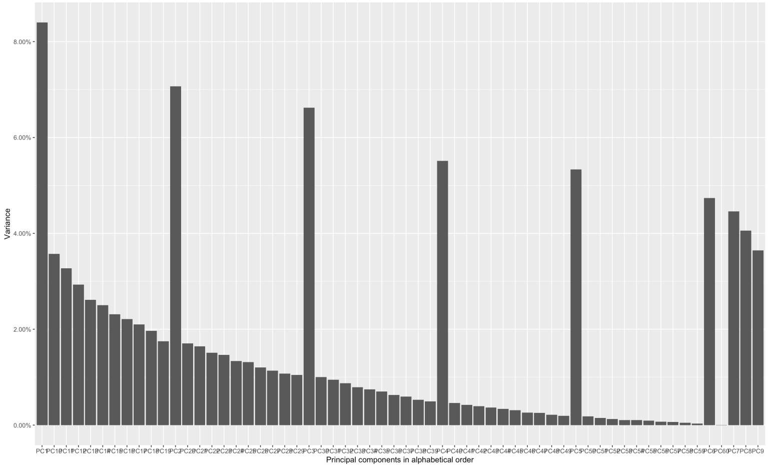
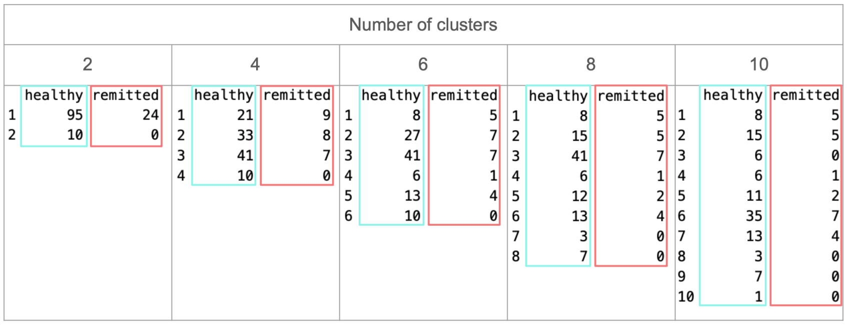


Figure 5.0.2: 30% dataset PCA scree plot

This PCA is a bad representation of the data for the same reasons that 4.1.1 is. For example, the first principal component accounts for less than 8.25% of the variance. This is the case for the PCA conducted on the 50% dataset and that conducted on the 80% dataset as well. The trends in the subject data are reflected in the smaller datasets. This also holds true for hierarchical clustering, k-means, and KNN.



| Number of clusters | | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | | | 4 | | | 6 | | | 8 | | | 10 | | |
| | healthy | remitted | | healthy | remitted | | healthy | remitted | | healthy | remitted | | healthy | remitted |
| 1 | 95 | 24 | 1 | 21 | 9 | 1 | 8 | 5 | 1 | 8 | 5 | 1 | 8 | 5 |
| 2 | 10 | 0 | 2 | 33 | 8 | 2 | 27 | 7 | 2 | 15 | 5 | 2 | 15 | 5 |
| | | | 3 | 41 | 7 | 3 | 41 | 7 | 3 | 41 | 7 | 3 | 6 | 0 |
| | | | 4 | 10 | 0 | 4 | 6 | 1 | 4 | 6 | 1 | 4 | 6 | 1 |
| | | | | | | 5 | 13 | 4 | 5 | 12 | 2 | 5 | 11 | 2 |
| | | | | | | 6 | 10 | 0 | 6 | 13 | 4 | 6 | 35 | 7 |
| | | | | | | | | | 7 | 3 | 0 | 7 | 13 | 4 |
| | | | | | | | | | 8 | 7 | 0 | 8 | 3 | 0 |
| | | | | | | | | | | | | 9 | 7 | 0 |
| | | | | | | | | | | | | 10 | 1 | 0 |

Figure 5.0.3: 50% dataset hierarchical clustering

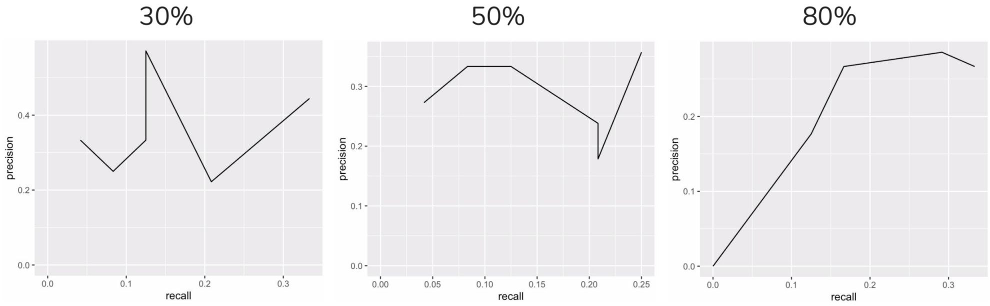


Figure 5.0.4: Selected feature datasets' precision-recall curves

The hierarchical clustering carried out on the 30% dataset, displayed in figure 5.0.3, displays the same behavior that the hierarchical clustering on the subject data (figure 4.2.2) does. Similarly, the precision-recall curves for all three selected feature datasets, displayed in figure 5.0.4, show that the KNN models constructed from selected features do not predict with significantly greater accuracy than those constructed from the whole dataset (subject data). For example, one KNN model from the 30% dataset predicts with greater precision than the best subject data model (over 50% compared to approximately 30% for the subject data) but at the cost of greatly reduced recall (less than 15% compared to approximately 40% for the subject data). None of the models can be considered accurate predictors.

# Chapter 6

# Conclusions and Next Steps

The results suggest that network metrics derived from dynamic signatures inferred from fMRI data are not sufficient to consistently classify between healthy and remitted subjects on their own. They also may indicate that connectivity patterns concerning the amygdala-hippocampal complex do not differ significantly between healthy and remitted subjects. However, the remitted patient sample was far smaller than the healthy subject pool, which may have biased the results. Thus, further research would ideally involve a larger number of healthy subjects and subjects who, at the time of data collection, were diagnosed with MDD. It would also include a wider range of machine learning and feature selection techniques, as well as motif searches in the whole networks.

# Bibliography

[1] Major depression, Feb 2019.

[2] Kuljeet Singh Anand and Vikas Dhikav. Hippocampus in health and disease : An overview. *Annals of Indian Academy of Neurology*, 2017.

[3] J. P. Hamilton, M. Siemer, and I. H. Gotlib. Amygdala volume in major depressive disorder: A meta-analysis of magnetic resonance imaging studies, 2008.

[4] Gareth James, Daniela Witten, Trevor Hastie, and Robert Tibshirani. *An introduction to statistical learning*, volume 112. Springer, 2013.

[5] Jacques Lemelin, Steve Hotz, Robert Swensen, and Thomas Elmslie. Depression in primary care. why do we miss the diagnosis? *Canadian Family Physician*, 40:104, 1994.

[6] Sayak Paul. Beginner's guide to feature selection in python, Sep 2018.

[7] Elizabeth A Phelps. Human emotion and memory: interactions of the amygdala and hippocampal complex. *Current opinion in neurobiology*, 14(2):198–202, 2004.