STATISTICS, OPTIMIZATION AND INFORMATION COMPUTING
Stat., Optim. Inf. Comput., Vol. **12**, January 2024, pp 15–33.
Published online in International Academic Press (www.IAPress.org)

Identifying the Neurocognitive Difference Between Two Groups Using Supervised Learning

Ramchandra Rimal

Department of Mathematical Sciences, Middle Tennessee State University, USA

Abstract
Brain Imaging Analysis is a dynamic and exciting field within neuroscience. This study is conducted with two main objectives. First, to develop a classification framework to enhance predictive performance, and second, to conduct a comparative analysis of accuracy versus inference using brain imaging data. The dataset of chess masters and chess novices is utilized to identify neurocognitive differences between the two groups, based on their resting-state functional magnetic resonance imaging data. A network of connections between brain regions is created and analyzed. Standard statistical learning techniques and machine learning models are then applied to distinguish connectivity patterns between the groups. The trade-off between model precision and interpretability is also assessed. Finally, model performance measures, including accuracy, sensitivity, specificity, and AUC, are reported to demonstrate the effectiveness of the model framework.

Keywords brain imaging, fMRI data, PCA, supervised learning, LSTM, GRU

DOI: 10.19139/soic-2310-5070-1340

1. Introduction

Recent advancements in machine learning and computational capabilities paved the way to discover the complex relationship between neuronal activity inside the human brain. Investigating the brain's functional organization and how it varies from the disease or continuous training is of high importance among researchers, medical practitioners, and policymakers [1, 2]. Resting-state functional magnetic resonance imaging (rs-fMRI) has great potential to advance our understanding and help identify these phenomena. Identifying these biomarkers has far-reaching implications such as helping treatment decisions, aiding in diagnosis, identifying personality traits, and developing new therapies for improving the brain's cognitive ability [1]. The most common ways machine learning methods study the rs-fMRI data are supervised learning and unsupervised learning. The unsupervised learning methods primarily focused on the functional organization of the brain dynamics. On the other hand, supervised learning uses functional connectivity to make individual-level predictions where functional connectivity measures the statistical relationship between blood oxygenated level-dependent (BOLD) fluctuations among spatially distributed brain regions [1, 2]. The analysis is performed by analyzing the resting-state functional connectivity between the brain regions of interest.

Imaging studies unfold insights about normal brain function and structure, neural processing and neuroanatomic expressions of psychiatric and neurological disorders, and changes in neural activity associated with treatment response [2]. Among the several widely used imaging modalities, functional magnetic resonance imaging (fMRI) is one of the most popular techniques. fMRI is popular because it is a non-invasive method to obtain neural

*Correspondence to: Ramchandra Rimal (Email: ramchandra.rimal@mtsu.edu). Department of Mathematical Sciences, Middle Tennessee State University. 612 Old Main Cir KOM, Murfreesboro, TN, 37132 USA .

ISSN 2310-5070 (online) ISSN 2311-004X (print)

activity information from the human brain. The fMRI quantifies brain activity by measuring correlates of blood flow and metabolism. In particular, activation of the brain is associated with a localized increase in oxygen consumption. The increase in oxygen consumption leads to the oxygen-rich blood flow toward the active area to meet the increased demand. The increased supply of oxygen dominates the increased demand and leads to a relative increase in oxyhemoglobin and a decrease in deoxyhemoglobin in activated brain areas. The common form of fMRI works by leveraging magnetic susceptibility properties of hemoglobin in capillary red blood cells, which delivers oxygen to neurons. Since hemoglobin is diamagnetic when oxygenated and paramagnetic when deoxygenated, the MR scanner produces blood oxygenated level-dependent (BOLD) signals that vary according to the degree of oxygenation [2]. Due to this reason, fMRI can be used to create the distribution maps of localized brain activity. There are two different ways the fMRI is taken, the task-based fMRI and the rs-fMRI. Although the person is not doing any task during the rs-fMRI, spontaneous fluctuations in the resting brain are prevalent whose causal origin is largely unknown. Several studies have been done to explore whether resting-state coherent fluctuations have a neuronal origin or are just due to physiological artifacts [3, 4, 5, 6]. Those research found support for the neuronal basis of BOLD-based resting-state functional connectivity. This study analyzes the resting-state functional connectivity to discriminate the neurocognitive difference between two healthy groups known as chess masters and the chess novices and identify the most discriminating brain regions using Automated Anatomical Labeling (AAL) atlas [7].

Several research works based on resting-state functional connectivity (RSFC) have been done to classify healthy and unhealthy people. In addition, rs-fMRI has been used to identify the neurocognitive difference between the groups of people, for example, the cognitive difference between people with schizophrenia and healthy patients [8, 9] studied the functional connectivity of the group of an aging population. The authors observed that the difference emerges in the intrinsic network architecture as associations changes between network interactivity, cognitive functioning, and age. They found that connectivity varied by cognition and age for within- and between-network connections. The relationship between functional connectivity in large-scale brain networks and age over the adult lifespan was studied in [10]. Their results showed significant alterations in the functional connectivity patterns characterized by weaker within-network connectivity and higher between-network connectivity in advanced age. Khan et al. (2020) [11] implemented the sequential forward feature selection approach to obtain the most discriminating features between autistic and healthy controls. The results in [12] show a decreased functional connectivity among the autistic subjects as compared to the controlled subjects. In the task unrelated neuronal activity between 23 autistic and 20 control subjects, the study in [13] found disruptions in functional connectivity. The work on [8] implemented linear classification techniques to classify Alzheimer's and Schizophrenia by analyzing fMRI data. [14] implemented several classification techniques to classify autistic and healthy patients and concluded that the neural networks outperformed other methods. [15] utilized the MRI scans of chess masters and novices to estimate morphometric connectivity. The authors introduced a novel functional morphometric similarity connectome by integrating functional and anatomical features, followed by classification using the SVM model after the feature selection technique is implemented. In this work, we explore the neurocognitive difference of two healthy groups, one is trained on the specific task that requires intensive use of the human brain, and the other does not, using statistical and machine learning techniques. Given that chess involves significant cognitive engagement, we examine how variations in cognitive traits can be forecasted through disparities in resting-state functional connectivity between adept chess players and novice chess players.

In a nutshell, plenty of research has been done in brain imaging analysis to identify the neurocognitive differences between healthy and unhealthy groups. Some research focuses on complex statistical or machine learning techniques that focus on achieving higher accuracy while others only use the standard statistical techniques for better statistical inference [16, 2, 1, 17]. In this work, we analyze the rs-fMRI data of two healthy groups of chess masters and chess novices and perform a comparative study of the performance among simple and complex models. We implemented the logistic regression (LR), support vector classifier (SVC), random forest classifier (RFC), gradient boosting classifier (GBC), Long short-term memory (LSTM) neural network, and the gated recurrent unit(GRU) neural network. The former models are tentatively more interpretable and less complex than the latter.

The implemented state of art deep learning models such as LSTM and GRU are well-suited for capturing complex relationships. On the other hand, the implemented standard machine learning models are well suited for statistical inference. The paper discusses the results of optimal models in both improved prediction and inference settings. Additionally, this work contributes to a limited body of research focused on identifying neurocognitive differences among healthy groups.

2. METHODS

The remainder of the paper is structured as follows: We start by providing a concise overview of the classification methods employed for the analysis in Section 2.2. Subsequently, the process of data collection and preprocessing is elaborated upon in Section 2.3. The techniques for establishing functional connectivity among brain regions of interest are introduced in Section 2.4. We detail the application of principal component analysis (PCA) for dimension reduction in Section 2.5. Moving forward, the experimental methods and model performance are scrutinized in Section 3.1 and Section 3.3 respectively. The comparison between model accuracy and their utility in statistical inference is outlined in Section 3.4. Finally, we conclude with the presentation of our results and discussions in Section 4. The references cited in this article can be found in the references section at the end.

2.1. Computational Framework

The proposed model framework is visualized in Figure 1. The research starts by extracting time series data from the AAL atlas, as proposed in [7]. Subsequently, RSFCs are created using Pearson's correlation, Spearman's rank correlation, and precision matrix. From these connectivity matrices, we extract the upper triangular elements to serve as features. To reduce dimensionality, we employ the PCA technique, resulting in 40 features for the Pearson RSFC and Spearman's RSFC data of 54 subjects, while the Precision RSFC data consists of 20 features. These selected features are then utilized for modeling. We construct classification models for all experimented classification methods, optimizing hyperparameters to achieve the best performance. With the chosen hyperparameters, the final model is trained on the training data and applied to classify chess masters and novices in the test dataset. To evaluate our model's performance, we use various measures, including accuracy, sensitivity, specificity, and AUC. Detailed procedures can be found in the corresponding sections and subsections.

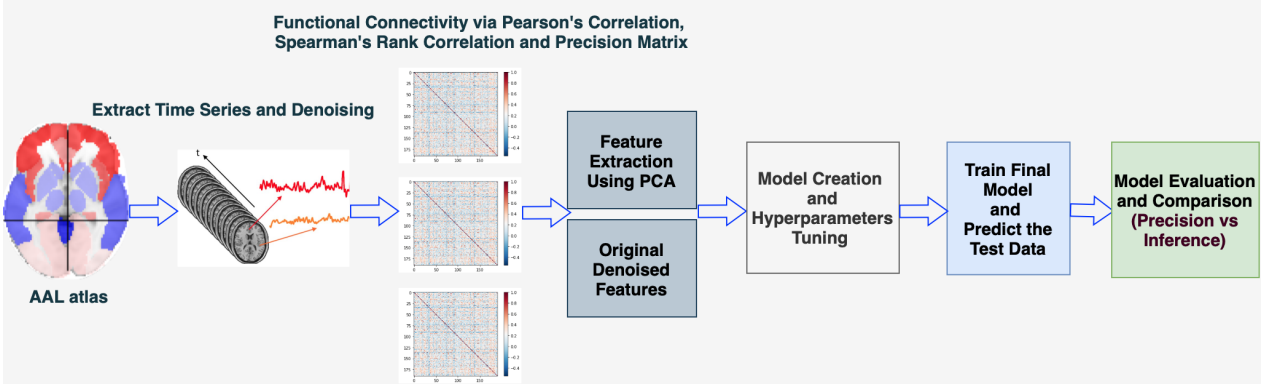


Figure 1. **Schematic diagram of proposed model framework**

2.2. Review of Classification Techniques

Logistic Regression

Logistic regression for a binary classification models the probability that the response variable Y belongs to a particular category. In this case, the chess masters (coded 1) or chess novices (coded 0). Like linear regression,

logistic regression incorporates both intercept and slope parameters. However, because probabilities must lie between 0 and 1, a straightforward intercept and slope model does not ensure this constraint. Instead, the goal is to model $p(X) = \Pr(Y = 1|X)$ using a function that generates outputs between 0 and 1 across all predictor variable values X [18, 19].

When $p(X) = p$ signifies event probability, the odds of the event are given by $\frac{p}{1-p}$. These odds span values from 0 to ∞. Logistic regression models the log odds of the event through a linear function as follows:

$$\log\left(\frac{p}{1-p}\right) = \beta_0 + \beta_1 X$$

Upon straightforward manipulation, we arrive at the logistic function:

$$p(X) = \frac{e^{\beta_0+\beta_1 X}}{1 + e^{\beta_0+\beta_1 X}}$$

To estimate the parameters β_0 and β_1, we utilize training data. New observations are then classified based on the logistic function's output. The relevant scientific studies that have applied logistic regression to classify fMRI data includes works such as [20, 21, 22].

Support Vector Classifier

The Support Vector Classifier (SVC), also referred to as the Maximum Margin Classifier, works to maximize the separation distance between the classification boundary and the nearest training data point. In the context of our study, we are categorizing individuals as either chess masters (coded as 1) or chess novices (coded as -1) for simplicity. In the case of a linear SVC, the decision boundary is linear.

To classify a new observation $\mathbf{u}$, we employ the decision function:

$$D(u) = \beta_0 + \sum_{i=1}^{n} y_i \alpha_i \mathbf{x_i}'\mathbf{u}$$

In this equation, $\mathbf{x_i}$ represents predictor data, and the coefficients $\alpha_i \geq 0$ are the parameters. The predictors $\mathbf{x_i}$ with non-zero alpha values are termed as support vectors ([18, 23, 19]). The decision function is solely dependent on these support vectors. When $D(u) > 0$, the sample is assigned to the ChessMasters class; otherwise, it's assigned to the novice class. The concept for the nonlinear SVC is analogous to the linear SVC, with the key distinction being the use of a nonlinear decision boundary. Instead of the linear $\mathbf{x_i}'\mathbf{u}$ term in the decision function, a kernel function $K(\mathbf{x_i}, \mathbf{u})$ is utilized. Numerous studies (e.g., [24, 25, 26, 27]) have implemented SVC with success for the classification of fMRI data.

Random Forest Classifier

The random forest technique is an ensemble method that combines outcomes from multiple individual decision trees. Suppose we have a dataset with n observations and p predictors. During the process of constructing a decision tree, a random subset of m predictors out of the p available predictors is selected as candidates for splits whenever a split within a tree is being evaluated. Each time a split occurs, only one of these m predictors is utilized from the entire set of p predictors. A fresh subset of m predictors is chosen for each split. Typically, m is chosen to be approximately $\sqrt{p}$, meaning that the number of predictors considered for each split is roughly equal to the square root of the total number of predictors. The another common choice of m is $\log(p)$. In this methodology, we generate B distinct bootstrapped training datasets. Each of these datasets is used to train the method, predicting out-of-bag samples, and ultimately aggregating all the predictions. For a new test observation, classification is achieved by recording the predicted class from each of the B trees and determining the majority vote. The final prediction corresponds to the most frequently occurring majority class among the B predictions. Numerous research studies in the past employed the random forest approach for the classification of fMRI data [28, 29, 30].

Gradient Boosting

Gradient Boosting is an ensemble approach where multiple weak classifiers are aggregated to construct a robust classifier. This technique begins by assigning initial weights to each sample. During each iteration, the algorithm identifies the optimal classifier based on the current sample weights. Samples that are misclassified in the current round receive increased weights in the subsequent iteration, while correctly classified samples are assigned reduced weights. Essentially, the algorithm assigns progressively higher weights to challenging-to-classify samples until the model effectively classifies them. A stage weight is calculated according to the error rate at each iteration. Finally, the sequence of weighted classifiers is merged into an ensemble, offering enhanced potential for accurate classification compared to individual weak classifiers [31].

For a binary classification task, where one class is labeled +1 and the other -1, we begin by generating k distinct bootstrap training datasets. Our approach is trained on the i-th bootstrap training set, followed by prediction on the out-of-bag samples. In each iteration, the stage weights are adjusted as detailed above. As soon as the model achieves accurate classification for all samples, the weighted average of all predictions is computed. The new test observation will be classified to +1 class if the weighted average obtained is positive otherwise it will be classified to -1 class. Some of the recent works that utilized this method for classification problems associated with brain imaging analysis are listed here [32, 33, 34, 35].

A brief overview of LSTM

The LSTM model is a well-known deep learning technique used for sequence classification tasks. Traditional recurrent neural networks (RNNs) face challenges in learning long-term dependencies due to the vanishing gradient problem [36], which the LSTM model was designed to address. LSTM employs memory cells to counter the vanishing gradient issue, comprising input, hidden, cell state, and output layers [37, 38, 39]. A vital element in the LSTM architecture is the cell state which runs through the chain that flows linearly, maintaining information continuity. The gate mechanism of LSTM modifies or removes cell state information, allowing selective passage through layers involving sigmoid and hyperbolic tangent activations, as well as point-wise multiplication.

The architecture of LSTM at time t is presented in the Fig 2. In this figure the four gates —output, change, input, and forget —are shown with their operations at time t.

For a given input sequence $\{x_1, x_2, \ldots, x_n\}$, $x_t \in \mathbb{R}^{k\times 1}$ is the input sequence at time t. The memory cell c_t updates the information using three gates: input gate i_t, forget gate f_t, and change gate $\tilde{c}_t$. The hidden state h_t is updated using output gate o_t and the memory cell c_t. The respective gates and layers compute the following functions at time t:

$$
\begin{aligned}
i_t &= \sigma(W_i x_t + W_{hi} h_{t-1} + b_i),\\
f_t &= \sigma(W_f x_t + W_{hf} h_{t-1} + b_f),\\
o_t &= \sigma(W_o x_t + W_{ho} h_{t-1} + b_o),\\
\tilde{c_t} &= \tanh(W_c x_t + W_{hc} h_{t-1} + b_c),\\
c_t &= f_t \otimes c_{t-1} + i_t \otimes \tilde{c_t},\\
h_t &= o_t \otimes \tanh(c_t)
\end{aligned}
$$

where, σ represent the sigmoid function. The operator $\otimes$ denotes element-wise multiplication. $W \in \mathbb{R}^{d\times k}$ and $W_h \in \mathbb{R}^{d\times d}$ are weight matrices, while $b \in \mathbb{R}^{d\times 1}$ represents bias vectors. The variables n, k, and d correspond to sequence length, number of features, and hidden size respectively [41, 42, 43]. Recent literature [44, 45, 46, 47, 48] has shown the utilization of LSTM neural networks for fMRI data classification.

To match LSTM architecture's dimensionality requirements, the input sequence X_t is formed by creating m continuous sequences $x_t : x_{t+m-1}$, resulting in a $k \times m$ matrix for $t \in \{1, 2, \ldots, n-m-1\}$. The LSTM output h_t serves as the feature representation for the input sequence X_t at time t, denoted as $h_t = \text{LSTM}(X_t, h_{t-1}, c_{t-1}, w)$ where w represents learnable parameters. As the final hidden state h_f contains key information from the input sequence, it is transformed into a vector through a dense layer.

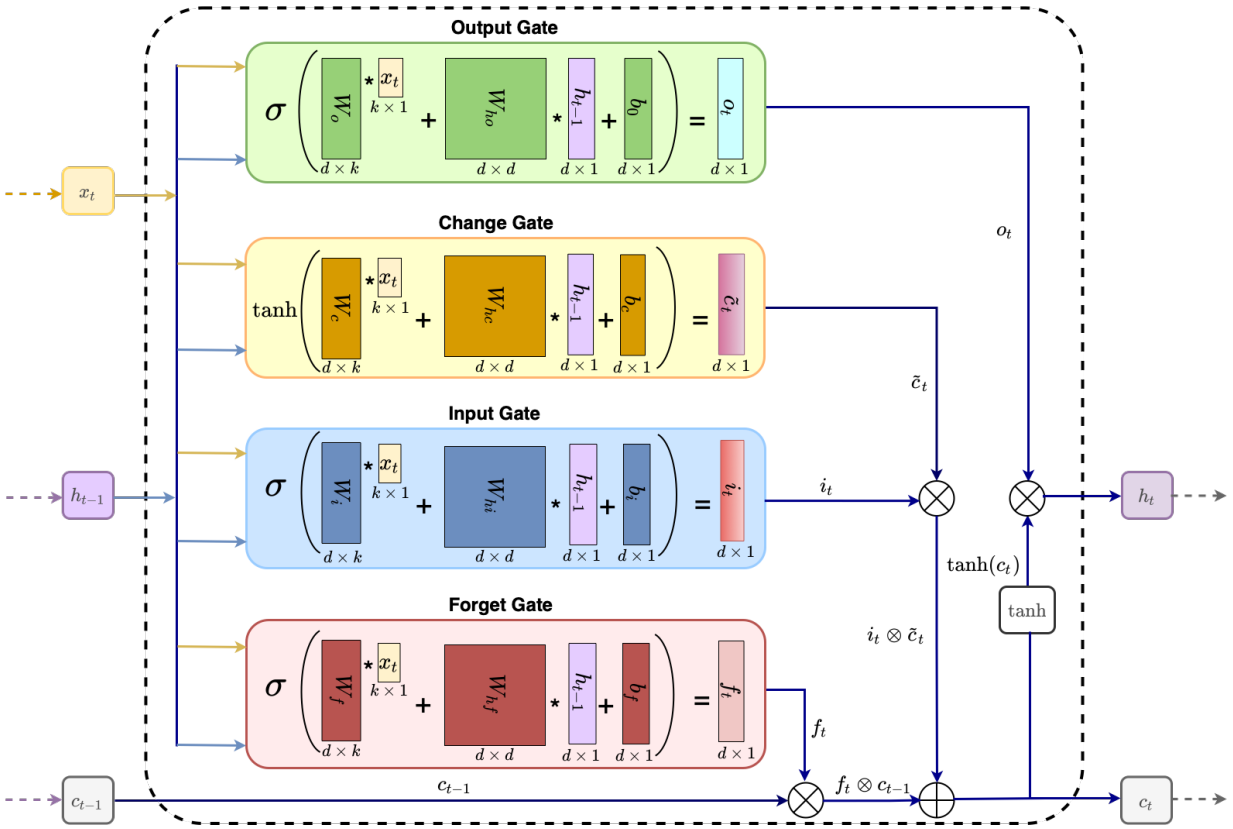


Figure 2. **LSTM architecture [40, 36].**

A brief overview of GRU

The GRU, proposed by [49], is a more recent model that simplifies the LSTM's four gates into three: the update, change, and reset gates. The update gate combines the functions of LSTM's input and forget gates. GRU is computationally efficient, requiring fewer parameters for estimation compared to the LSTM model (Dey et al., 2017; Chung et al., 2014). The architecture of the GRU at time t is presented in Figure 3, illustrating the operations of the reset, change, and update gates at time t. Similar to the LSTM model, for a given input sequence $x_1, x_2, \ldots, x_n$, with $x_t \in \mathbb{R}^{k\times 1}$ representing the input sequence at time t. The GRU model's gates and layers compute the following functions at time t:

$$
\begin{aligned}
r_t &= \sigma(W_r x_t + W_{hr} h_{t-1} + b_r),\\
u_t &= \sigma(W_u x_t + W_{hu} h_{t-1} + b_u),\\
\tilde{c}_t &= \tanh(W_c x_t + W_c(r_t \otimes h_{t-1}) + b_c),\\
h_t &= (1-u_t)\otimes h_{t-1} + u_t \otimes \tilde{c}_t
\end{aligned}
$$

where, σ represents the sigmoid function, $\otimes$ is the element-wise product; $W \in \mathbb{R}^{d\times k}$ and $W_h \in \mathbb{R}^{d\times d}$ are weight matrices, while $b \in \mathbb{R}^{d\times 1}$ serves as bias vectors. Additionally, n, k, and d denote the sequence length, the number of features, and the hidden size respectively [51]. Similar to the LSTM model, the input sequence X_t for the GRU model is formulated. The output h_t of the GRU denotes a feature representation for input sequence X_t at time t, expressed mathematically as:

$$h_t = GRU(X_t, h_{t-1}, w),$$

where w encompasses all parameters that can be learned through model training.

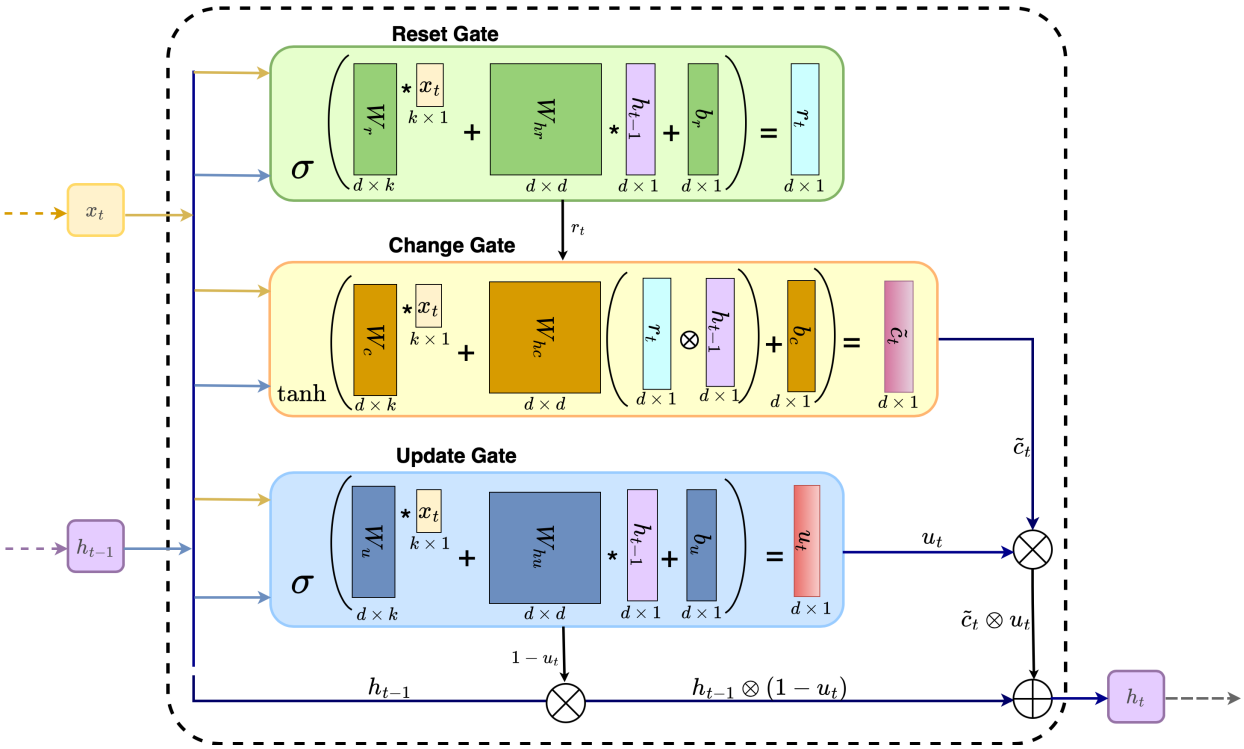


Figure 3. **GRU architecture [50, 49].**

2.3. Data Description and Preprocessing

We obtain the resting-state fMRI data from the Neuroimaging Tools and Resources Collaboratory. The dataset was made available by the Huaxi MR Research Center of Sichuan University. The dataset includes fMRI scans from 29 professional chess players and 29 age and sex-matched novices. Data collection utilized a Siemens 3T TRIO system at the West China Hospital of Sichuan University, Chengdu, China, employing an EPI sequence with parameters TR=2000 ms, TE=30 ms, Bandwidth=2442 Hz/Px, Flip angle=90 degrees, FOV=240 $\times$ 240 mm^2, and slice thickness=5 mm. Each volume consisted of thirty interleaved 64 $\times$ 64 slices. For more information about data and the protocol, please refer to the following article [52]. The chess masters dataset is preprocessed using the data processing assistant for resting-state fMRI *(DPARSF 4.1)* framework [53]. A total of 54 observations, 27 on Chess Masters and 27 on Chess Novices, were chosen for analysis. The preprocessed fMRI data for each subject constitute a matrix of dimensions 116 $\times$ 200, with rows representing brain regions of interest and columns depicting time points. These 116 brain regions are sourced from the AAL atlas. Figure 4 provides a visualization of the 116 brain regions as per the AAL atlas.

Denoising was performed using adaptive wavelet threshold denoising with Bayesian estimation [54], employing Haar wavelet and soft thresholding from the scikit-image package [55]. Wavelet-based denoising methods with limited smoothing effectively preserve image sharpness and active region shapes. The adaptive wavelet threshold denoising technique, employing Bayesian estimation, differentiates between signal and noise components to maintain frequency information and minimize noise impact. This method proves superior to the original Gaussian filter in retaining useful signals [56, 57, 58]. Both denoised and original data were used for experimentation and comparison.

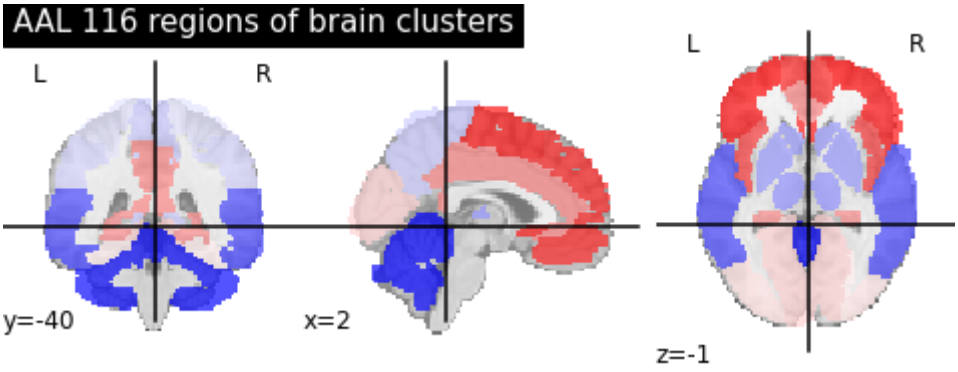


Figure 4. Visualization of 116 brain regions

2.4. Generation of Functional Connectivity

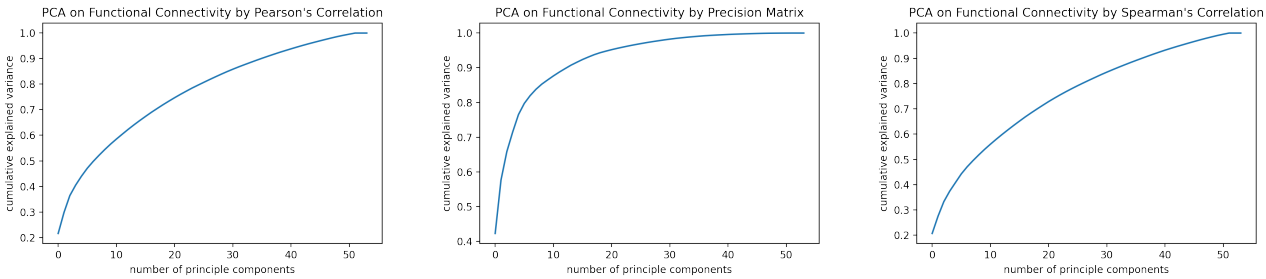
The network of resting-state functional connectivity (RSFC) is constructed using both denoised and original data. Three methods, Pearson's correlation, Spearman's rank correlation, and precision matrix are employed to create the connectivity matrix. Initially, Pearson's correlation coefficient is chosen due to its common usage in literature [59, 60]. However, partial correlation is deemed superior for direct connectivity assessment, considering potential influences from other regions [61, 62]. Therefore, graphical lasso [63] is utilized to compute the inverse covariance matrix as a connectivity matrix. Finally, Spearman's rank correlation [64] is employed for its ability to capture both linear and non-linear relationships between brain regions. The utilization of these three metrics aims to compare their impact on model performance within the proposed framework. The connectivity matrices being symmetric (mxm), upper diagonal elements form a feature vector of length $\frac{m*(m-1)}{2}$ due to self-connections on the diagonals being omitted. Each feature symbolizes the correlation between two brain regions conditioned on others. The dataset consists of 54 rows and 6670 columns, with rows representing subjects and columns representing features.

2.5. Dimension Reduction using Principal Component Analysis

We possess 54 observations, each with 6670 features, leading to a high-dimensional statistics-related challenge where the feature count significantly surpasses the sample size. To address this, Principal Component Analysis (PCA) is employed, mitigating common issues linked with high-dimensional data, such as computational complexity and inflated error rates due to multiple testing corrections during outcome-associated feature testing. PCA simplifies high-dimensional data complexity by transforming it into fewer dimensions that serve as feature summaries [65, 66]. Although ICA and its extensions are widely used in the rs-fMRI field, their modeling of linear representations for non-Gaussian data raises concerns about whether such transformations effectively capture the relationship between independent latent sources and fMRI data. Additionally, ICA fails to assign importance to its components, preventing the differentiation between strong and weak sources [1, 67].

After the PCA was implemented, the numbers of principal components were chosen so that they could explain the data variability of at least 90 percent. The 40 principal components that correspond to Pearson RSFC and Spearman's RSFC were chosen, and for Precision RSFC, 20 principal components were selected. Now each subject of the Pearson RSFC and Spearman's RSFC data has 40 features, and Precision RSFC data consists of 20 features, each feature representing the principal components. We have observations on the 54 subjects for the analysis. In this case, our data contains 54 rows representing the number of subjects, and the number of columns(40 or 20) equals the chosen number of principal components for each RSFC.

Additionally, we employed the feature selection methods known as SelectFromModel (SFM) and recursive feature elimination (RFE) technique, as described by [16]. The RFE technique begins with all features and iteratively eliminates the least significant feature until the desired feature count is achieved. In our analysis, the



(a) (b) (c)

Figure 5. The percentage of the variance explained by the number of principal components on the features obtained by (a) Pearson's Correlation, (b) Precision Matrix, and (c) Spearman's Correlation.

RFE method reduces the feature set to half of the total features at each iteration. The SelectFromModel is a versatile meta-transformer compatible with various estimators that assign feature importance. It discards features with importance values below a specified threshold parameter. It also provides built-in heuristics like "mean" and "median" and supports a feature limit using the max_features parameter. In our implementation, we utilized SFM with its default settings.

3. Experiments and Results

3.1. Experimental Methods

In this study, we explored supervised learning methods, including LR, LSVC, KSVC, RFC, GBC, LSTM, and GRU neural networks, for the classification of chess masters and novices. Implementation of these methods utilized scikit-learn and scikit-image libraries [68, 69] and TensorFlow [70, 71] in Python. The input for these models consists of feature vectors obtained through three different methods for constructing feature sets (FC). Our experimentation involved four distinct datasets: features extracted from the original data with and without PCA, as well as features from denoised data with and without PCA. Initially, five-fold cross-validation was performed on the entire dataset to determine the optimal model parameters. Subsequently, repeated ten-fold cross-validation was conducted using these parameters, with a performance evaluation on hold-out folds. The experiment was repeated ten times to ensure stable performance.

Data preparation steps were undertaken to ready the input data for LSTM and GRU models. The architecture for these models is illustrated in Figure 2 and Figure 3. The input sequence was constructed using $k = 6670$ features (reduced to $k = 40$ or $k = 20$ after PCA dimension reduction) with a time step of 1. LSTM model processed the input sequence along with memory cells, while the GRU model operated without memory cells. The transformed data, initially two-dimensional, was adjusted to a three-dimensional format (number of observations, time step, number of predictors) required by the models. The dataset was randomly divided into an 80% training set and a 20% test set.

For LSTM and GRU models, both single-layer and multi-layer architectures were experimented with. Hyperparameters such as learning rate, number of epochs, optimizers, and batch size were adjusted. Epochs were set to 50, and an early stopping criterion was employed to halt training after five consecutive instances of no validation loss improvement. Varying learning rates (0.1 to 0.001), neuron counts (10 to 250) with their combinations from one to three layers, and optimizers (Adam, RMSprop, Adadelta) were explored. Once optimal hyperparameters were identified, the model was trained using the chosen values, and performance was assessed on

the test set. To ensure reliable performance, each model was replicated ten times.

3.2. Hyperparameter Tuning

For LR, LSVC, KSVC, RFC, and GBC models, a five-fold cross-validation method was employed to determine the optimal hyperparameters. The LR model was tested with L_1, L_2, and elastic net penalties. However, the feature selection techniques did not enhance results for the lasso penalty implementation. The LR model's best performance came from the SFM feature selection method with a C value of 0.1 among the choices between 0.1 and 5.1, using the saga solver. LSVC utilized a C value of 0.3 from the range of 0.1 to 30, while KSVC with a linear kernel selected a C value of 2.2. For RFC, parameters like maximum depth (5 to 20), number of trees (500 to 2500), and impurity criterion (gini vs. entropy) were evaluated. The RFC model with 1000 trees, a maximum depth of 5, and the gini criterion was chosen. Gradient boosting employed a learning rate (0.1 to 0.001), number of trees (200 to 700), max depth (3 to 7), and various split quality criteria. The final model adopted a learning rate of 0.01, mean absolute error criterion, max depth of 7, and 500 estimators.

For LSTM and GRU models, data was split into training, validation, and test sets. Initially, 80% was for training and 20% for testing, then 20% of training became validation data. Hyperparameters included batch sizes, learning rates, optimizers, and epochs (set to 50 with early stopping). Optimizers were Adam, RMSprop, and Adadelta. Learning rates were chosen from 0.01, 0.05, 0.001, and 0.005. The batch sizes considered were 2, 4, and 8. Neuron counts ranged from 10 to 200 for single-layer and various multilayer combinations. For both LSTM and GRU, 12 models were evaluated with 36 combinations each, and the best-performing combination based on validation data was selected. This process was repeated for three data models: Spearman RSFC, Pearson RSFC, and Precision RSFC.

Model reliability was enhanced through ten replications, with average accuracy scores guiding hyperparameter selection. The top-performing LSTM and GRU models were based on Pearson's RSFC. Optimal settings for LSTM were a batch size of 4, a learning rate of 0.005, and a single-layer model with 200 neurons. Similarly, GRU model favored a batch size of 4, a learning rate of 0.005, and a three-layer configuration with neuron counts [70, 40, 10]. These settings were employed to train the models, which were then evaluated using the entire training dataset and tested on the test dataset for performance assessment.

3.3. Model Performance

The developed framework is tested on classifying chess masters and chess novices. The selected value of parameters obtained from the grid-search cross-validation has been used in the repeated ten-fold cross-validation for the LR, LSVC, KSVC, RFC, and GBC models. The ten-fold cross-validation is replicated ten times, and the average accuracy, sensitivity, specificity, and AUC scores obtained from the hold-out fold are reported. In addition, the average performance scores from these methods implementing the RFE and SFM techniques are recorded. The feature selection technique slightly improves the accuracy in comparison without using model selection.

For the LSTM and GRU models, various single-layer and multi-layers models has been implemented with best values of the hyperparameters chosen. However, we noticed the overfitting issue with these models. To fix the issue, drop-out strategy has been implemented. After implementation of drop out, the LSTM model outperforms all other tested models with significant improvement in some performance measures while GRU has better performance in other measures however not statistically significant different. As we worked with three different datasets corresponding to three different functional connectivity, the best performance score was obtained from the features using Pearson's connectivity with the following value of hyperparameters —200-neuron (LSTM) and [70, 40, 10] neuron (GRU), Adam optimizer with a learning rate of 0.005, and 50 epochs with early stopping and patience of 5.

The following tables summarize the performance of all implemented methods obtained by using Pearson's correlation, precision matrix, and Spearman's rank correlation as functional connectivity methods.

Table 1 displays the average performance scores of employed supervised learning models on PCA transformed Precision RSFCs. Our observations reveal that the KSVC model exhibits superior overall performance, with the GRU model emerging as its closest competitor among the array of compared methods. The top performance is seen

Table 1. **Performance scores of the various supervised learning models on Precision RSFC using PCA**

Metrics \Methods	KSVC	LSVC	LR +SFM	RFC	GBC	LSTM	GRU
Accuracy	**48.30**	45.53	46.47	47.4	45.83	46.67	48.18
Sensitivity	**69.49**	63.50	66.67	39.83	42.16	47.33	62.67
Specificity	28	29.67	31.17	**55.17**	50.83	46.11	36.11
AUC	**54.67**	36.61	51	41.55	44.39	43.22	53.44

from the KSVC model with notable scores in Accuracy (48.30), Sensitivity (69.49) and AUC (54.67). It's important to note that while the difference in accuracy and AUC scores is minimal, there's a noteworthy discrepancy in sensitivity and specificity. All other models shows varying scores and are inferior except for the specificity metrics. The RFC model demonstrates higher specificity, but at the expense of lower accuracy, sensitivity, and AUC. In comparison, the KSVC model demonstrates competitive predictive performance and computational efficiency, making it a favorable choice for this dataset with an additional advantage of its reduced computational demands compared to LSTM and GRU models..

Table 2. **Performance scores of various supervised learning models on PCA transformed Pearson's RSFC data**

Metrics \Methods	KSVC	LSVC	LR +SFM	RFC	GBC	LSTM	GRU
Accuracy	67.47	63.10	63.43	62.30	59.70	80.30	**80.91**
Sensitivity	66.83	56.67	65.50	63.83	61.33	76.67	**80**
Specificity	68.67	68.50	62.50	60.67	57.5	**83.33**	81.67
AUC	70.39	65.16	63.39	64.17	58.44	**84.72**	82.5

Table 2 displays the performance scores of various supervised learning models on PCA transformed Pearson's RSFC data. These scores underscore the consistent excellence of the LSTM and GRU models, which attain the highest marks across Accuracy (LSTM: 80.30, GRU: 80.91), Sensitivity (LSTM: 76.67, GRU: 80), Specificity (LSTM: 83.33, GRU: 81.67),and AUC (LSTM: 84.72, GRU: 82.50). In contrast, the LSVC, KSVC, LR + SFM, and RFC models exhibit varying levels of performance, highlighting the diversified outcomes among supervised learning approaches. To sum up, the LSTM and GRU models emerge as consistently robust choices for effective classification. Notably, the KSVC model also demonstrates relatively good performance metrics, despite its model simplicity and inferential advantage in comparison. Overall, the LSTM and GRU models display superior predictive performance without any statistically significant difference in PCA transformed Pearson's RSFC data.

Table 3. **Performance scores of various supervised learning models on Spearman's RSFC using PCA**

Metrics \Methods	KSVC	LSVC	LR +SFM	RFC	GBC	LSTM	GRU
Accuracy	64.93	5810	69.07	55.57	50.43	**76.06**	75.45
Sensitivity	61.33	52.67	64.83	57.17	52.67	69.33	**70**
Specificity	69.17	63.50	73	55	47.83	**81.67**	80
AUC	67.94	63.17	72.22	57.39	51.16	**81.44**	79.83

Table 3 reveals performance scores of diverse supervised learning models on Spearman's RSFC data transformed via PCA. Notably, the LSTM model stands out with superior overall performance, characterized by high accuracy (76.06), sensitivity (69.33), specificity (81.67), and AUC (81.44) scores. The GRU model closely follows with commendable performance in accuracy (75.45), sensitivity (70.00), specificity (80.00), and AUC (79.83) scores.

In comparison, the KSVC also demonstrates relatively good performance despite its simplicity. The other models have consistently shown lower performance across all metrics. Overall, the LSTM model emerges as the leader in predictive capabilities on this dataset.

From tables 1, 2, and 3, we can observe that the performance of the GRU model on Pearson's RSFC using PCA stands out as the best among all other compared methods. Although the difference with the LSTM is not statistically significant, the GRU model holds a slight computational advantage. This framework yields improved outcomes, likely attributed to the GRU's capacity to learn intricate predictor relationships, combined with Pearson's RSFC's capability to capture linear brain region connections.

3.4. Comparison of Results for Accuracy and Inference

In this section we discuss the results obtained from the two most promising simple model with the performance of the best model.

Table 4. **The average performance scores of the best model for accuracy and the best model for Inference**

Metrics	Best Model for Accuracy (GRU model on PCA transformed Pearson's RSFC)	Best Model for Inference (Kernel SVC model (linear Kernel) on Pearson's RSFC)
Accuracy	**80.91**	**60.80**
Sensitivity	**80**	**58.16**
Specificity	**81.67**	**63**
AUC	**82.50**	**65.61**

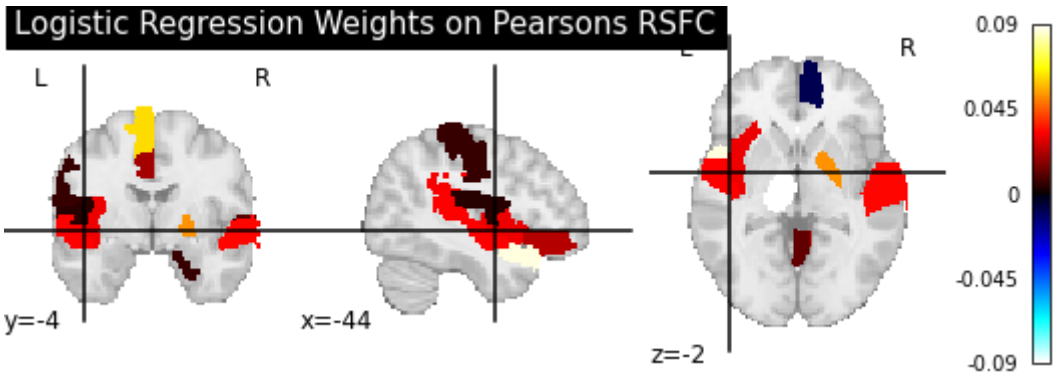


Figure 6. Visualization of the discriminating weights of the brain regions obtained by recursive feature elimination technique on the logistic regression model

In this section, we present the results derived from the two most promising models, optimized for accuracy and inference. The best model optimized for the accuracy is the GRU model on the PCA transformed Pearson's RSFC and the best model for inference is the Kernel SVC model (linear Kernel) on Pearson's RSFC data. Table 4 provides the average performance scores of the best model for accuracy and the best model for statistical inference. The GRU model with PCA on Pearson's RSFC data achieves the highest accuracy score of 80.91, while the Kernel SVC model (linear Kernel) on Pearson's RSFC data attains an accuracy score of 60.80. The GRU model also excels in sensitivity (80) and specificity (81.67), while the Kernel SVC model demonstrates a sensitivity of 58.16 and specificity of 63. Notably, the GRU model exhibits a specificity score of 81.67, outperforming the Kernel SVC's score of 63. Although the complex model offers significant improvements in accuracy and sensitivity, it compromises inference. The Kernel SVC model enables the identification of crucial brain regions contributing to the classification of chess masters and novices through feature importances while using the original features, whereas interpretability becomes challenging with PCA-transformed features.

The most influencing features among the logistic regression, linear support vector classifier and the support vector classifier with linear Kernel models were investigated. The discriminant weights of the logistic regression, SVC using the recursive feature selection method, and KSVC are given below. The logistic regression method selects the following regions of interest as shown in the figure 6.

The SVC method selects the following region as shown in the figure 7.

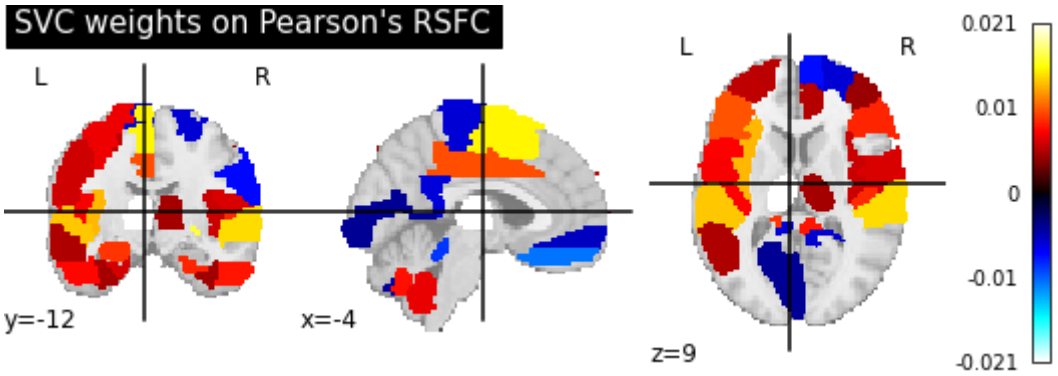


Figure 7. Visualization of the discriminating weights of the brain regions obtained by the recursive feature elimination technique on the support vector classification model

The KSVC method selects the following region as shown in the figure 8.

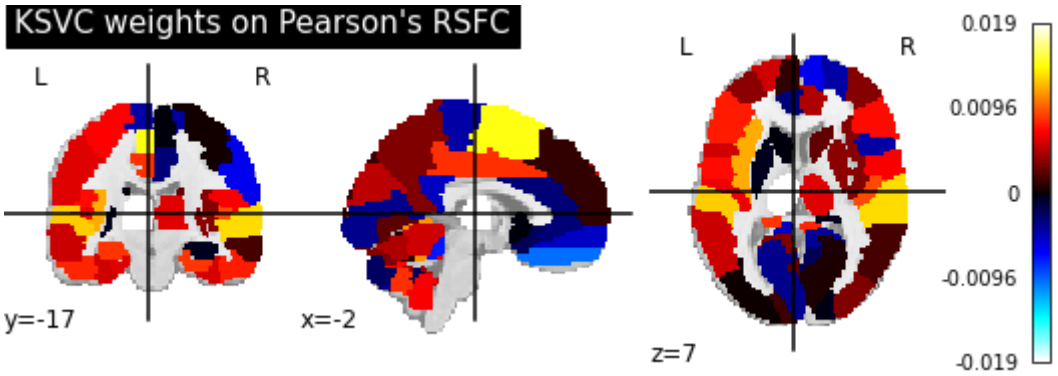


Figure 8. Visualization of the discriminating weights of the brain regions obtained by the support vector classifier with the linear kernel

We observed that the regions selected by the these methods are almost identical. Among the top five features selected by the two best-performing simple models, the four are the same, and the remaining one is different. From the agreement of these methods, we can say that these brain regions are most important in discriminating the two groups of interest.

Table 5. **Five most Influential Brain Regions selected by Logistic Regression on Pearson's Connectivity**

Rank	Brain Region Label	MNI Position		
		X	Y	Z
1	**Hippocampus_L**	-24	-2	-18
	Caudate_L	-12	10	8
2	Hippocampus_L	-24	-2	-18
	Caudate_R	14	11	8
3	Cerebelum_4_5_L	-31	-41	-22
	Cerebelum_Crus2_R	33	-70	-41
4	Cerebelum_Crus1_R	38	-68	-31
	Cerebelum_10_R	27	-35	-43
5	Vermis_3	-2	-41	-13
	Vermis_10	-3	-47	-33

Table 6. **Five most Influential Brain Regions selected by SVC on Pearson's Connectivity**

Rank	Brain Region Label	MNI Position		
		X	Y	Z
1	**Hippocampus_L**	-24	-2	-18
	Caudate_L	-12	10	8
2	Hippocampus_L	-24	-2	-18
	Caudate_R	14	11	8
3	Cerebelum_4_5_L	-31	-41	-22
	Cerebelum_Crus2_R	33	-70	-41
4	Cerebelum_Crus2_R	33	-70	-41
	Cerebelum_10_R	27	-35	-43
5	Vermis_3	-2	-41	-13
	Vermis_10	-3	-47	-33

Table 7. **Five most Influential Brain Regions selected by KSVC on Pearson's Connectivity**

Rank	Brain Region Label	MNI Position		
		X	Y	Z
1	Frontal_Sup_Orb_L	-17	46	-15
	Cerebelum_Crus2_R	33	-70	-41
2	**Hippocampus_L**	-24	-2	-18
	Caudate_L	-12	10	8
3	Hippocampus_L	-24	-2	-18
	Caudate_R	14	11	8
4	Cerebelum_4_5_L	-31	-41	-22
	Cerebelum_Crus2_R	33	-70	-41
5	Vermis_3	-2	-41	-13
	Vermis_10	-3	-47	-33

Table 5, 6, and 7 presents the top five influential brain regions selected by the LR LSVC and KSVC methods on identifying the chess masters and chess novices. Out of five selected regions, majority are the same just in a slightly

different order, and only one is different. All methods agree that the regions Hippocampus_L and Caudate_L are among the top two important regions. The most accurate model KSVC selects the regions Frontal_Sup_Orb_L and Cerebelum_Crus2_R as the most important region, whereas none of the other methods identify Frontal_Sup_Orb_L among the most important five regions. As there were very few results on this type of problem and even in a different setting so having a comparison is not possible [15, 27].

4. Discussion and Conclusion

In this article, we undertook the classification of chess masters and chess novices using a variety of supervised learning methods. This endeavor represents one of the few attempts, to the best of our knowledge, that aims to distinguish between two completely healthy groups through the analysis of fMRI data. The employed measures include RSFC derived from Pearson's correlation, Spearman's rank correlation, and the precision matrix obtained through graphical lasso. We compared the performance of classification models corresponding to each rs-RSFC measure, both on denoised and original data. The improvements in model performance remain relatively marginal with the use of denoised data and the results reported in the study corresponds to the denoised data. All the compared models performed poorly in the data with original features, among them, the best results obtained from the original features is discussed for the inference perspective. The inferior performance of overall models be attributed to the challenge posed by the curse of dimensionality, given the modest number of observations (54) in relation to the substantial number of features (6670). To mitigate this, we employed PCA for dimension reduction. While PCA enhanced model performance and significantly reduced computational time, it came at the expense of feature interpretability, particularly evident in classical machine learning models as the deep learning models has a known difficulty for statistical inference. The results on the PCA transformed three different RSFCs are thoroughly discussed.

On the top of several machine learning models such as LR, LSVC, KSVC, RFC, GBC, our exploration extended to LSTM and GRU models, encompassing a range of hyperparameter values. The most straightforward model with the highest accuracy was ultimately selected. Both LSTM and GRU models were equipped with a drop-out strategy to prevent overfitting. Comparatively, the GRU model applied to PCA-transformed Pearson's RSFC data exhibited superior performance, significantly enhancing accuracy and sensitivity. Meanwhile, LSTM models applied to the same data excelled in specificity and AUC scores. It's noteworthy that there is no statistically significant difference between the results of LSTM and GRU models; both outperform other models assessed in this study. On the other hand, the KSVC with linear kernel on the features obtained by Pearson's RSFC remains promising as it is simple enough for statistical inference and has an acceptable performance. We gain an insights on the important brain regions contributing to identify the differences between players. In our experiment, the KSVC with linear kernel emerges as the top competitor to the LSTM/GRU model among all the compared models capable of statistical inference.

Our work has some limitations. Since the sample size is small, the overall performance scores may depend on the split of the data set. In addition, the LSTM/GRU neural network expects to have a decent sample size to learn the value of the parameters. Since our sample size is small, the model performance is mediocre. Since this is a binary classification problem between healthy groups, there may not be a significant difference in the functional connectivity between the groups. From the analysis of performance metrics, it does not seem easy to classify these two groups. Exploring the new state of art attention based models, hybrid classification techniques, implementing various global optimizers and local optimizer in the model, applying other dimension reduction techniques such as independent component analysis, autoencoders and manifold learning methods is a matter of future work.

Acknowledgement

The author expresses gratitude to the anonymous reviewers for their valuable comments that enhanced the quality of the content presented in this article.

References

[1] Meenakshi Khosla et al. “Machine learning in resting-state fMRI analysis”. In: *Magnetic resonance imaging* 64 (2019), pp. 101–121.

[2] F. Bowman. “Brain Imaging Analysis.” In: *Annual review of statistics and its application* 1 (2014), pp. 61–8.

[3] R Cameron Craddock et al. “Disease state prediction from resting state functional connectivity”. In: *Magnetic Resonance in Medicine: An Official Journal of the International Society for Magnetic Resonance in Medicine* 62.6 (2009), pp. 1619–1628.

[4] MBCFDS De Luca et al. “fMRI resting state networks define distinct modes of long-distance interactions in the human brain”. In: *Neuroimage* 29.4 (2006), pp. 1359–1367.

[5] Christian F Beckmann et al. “Investigations into resting-state connectivity using independent component analysis”. In: *Philosophical Transactions of the Royal Society B: Biological Sciences* 360.1457 (2005), pp. 1001–1013.

[6] Jared A Nielsen et al. “Multisite functional connectivity MRI classification of autism: ABIDE results”. In: *Frontiers in human neuroscience* 7 (2013), p. 599.

[7] Nathalie Tzourio-Mazoyer et al. “Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain”. In: *Neuroimage* 15.1 (2002), pp. 273–289.

[8] Kamalaker Dadi et al. “Benchmarking functional connectome-based predictive models for resting-state fMRI”. In: *NeuroImage* 192 (2019), pp. 115–134.

[9] Margot D Sullivan et al. “Intrinsic neurocognitive network connectivity differences between normal aging and mild cognitive impairment are associated with cognitive status and age”. In: *Neurobiology of aging* 73 (2019), pp. 219–228.

[10] Epifanio Bagarinao et al. “Reorganization of brain networks and its association with general cognitive performance over the adult lifespan”. In: *Scientific reports* 9.1 (2019), pp. 1–15.

[11] Naseer Ahmed Khan et al. “A Three-Stage Teacher, Student Neural Networks and Sequential Feed Forward Selection-Based Feature Selection Approach for the Classification of Autism Spectrum Disorder”. In: *Brain sciences* 10.10 (2020), p. 754.

[12] Michal Assaf et al. “Abnormal functional connectivity of default mode sub-networks in autism spectrum disorder patients”. In: *Neuroimage* 53.1 (2010), pp. 247–256.

[13] Tyler B Jones et al. “Sources of group differences in functional connectivity: an investigation applied to autism spectrum disorder”. In: *Neuroimage* 49.1 (2010), pp. 401–414.

[14] Milan N. Parikh, Hailong Li, and Lili He. “Enhancing Diagnosis of Autism With Optimized Machine Learning Models and Personal Characteristic Data”. In: *Frontiers in Computational Neuroscience* 13 (2019), p. 9. ISSN: 1662-5188. DOI: 10.3389/fncom.2019.00009. URL: https://www.frontiersin.org/article/10.3389/fncom.2019.00009.

[15] Harish RaviPrakash et al. “Morphometric and functional brain connectivity differentiates chess masters from amateur players”. In: *Frontiers in Neuroscience* 15 (2021), p. 629478.

[16] Hariharan Ravishankar et al. “Recursive feature elimination for biomarker discovery in resting-state functional connectivity”. In: *2016 38th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*. IEEE. 2016, pp. 4071–4074.

[17] Md Rahman et al. “A Review of machine learning methods of feature selection and classification for autism spectrum disorder”. In: *Brain sciences* 10.12 (2020), p. 949.

[18] Jerome Friedman, Trevor Hastie, Robert Tibshirani, et al. *The elements of statistical learning*. Vol. 1. 10. Springer series in statistics New York, 2001.

[19] Gareth James et al. *An introduction to statistical learning*. Vol. 112. Springer, 2013.

[20] Frank de Vos et al. “A comprehensive analysis of resting state fMRI measures to classify individual patients with Alzheimer’s disease”. In: *Neuroimage* 167 (2018), pp. 62–72.

[21] Logan Grosenick, Stephanie Greer, and Brian Knutson. “Interpretable classifiers for FMRI improve prediction of purchases”. In: *IEEE transactions on neural systems and rehabilitation engineering* 16.6 (2008), pp. 539–548.

[22] Destie Provenzano et al. “Logistic Regression Algorithm Differentiates Gulf War Illness (GWI) Functional Magnetic Resonance Imaging (fMRI) Data from a Sedentary Control”. In: *Brain sciences* 10.5 (2020), p. 319.

[23] Max Kuhn, Kjell Johnson, et al. *Applied predictive modeling*. Vol. 26. Springer, 2013.

[24] Mengyue Wang et al. “Support vector machine for analyzing contributions of brain regions during task-state fMRI”. In: *Frontiers in neuroinformatics* 13 (2019), p. 10.

[25] Jiangfen Wu et al. “Resting state fMRI feature-based cerebral glioma grading by support vector machine”. In: *International journal of computer assisted radiology and surgery* 10.7 (2015), pp. 1167–1174.

[26] Xia-an Bi et al. “Random support vector machine cluster analysis of resting-state fMRI in Alzheimer’s disease”. In: *PloS one* 13.3 (2018), e0194479.

[27] Xin Yang, Ramchandra Rimal, and Tiffany Rogers. “Functional Connectivity Based Classification for Autism Spectrum Disorder Using Spearman’s Rank Correlation”. In: *2022 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES)*. IEEE. 2022, pp. 46–51.

[28] Xi Zhu et al. “Random forest based classification of alcohol dependence patients and healthy controls using resting state MRI”. In: *Neuroscience letters* 676 (2018), pp. 27–33.

[29] Maya A Reiter et al. “Performance of machine learning classification models of autism using resting-state fMRI is contingent on sample heterogeneity”. In: *Neural Computing and Applications* 33.8 (2021), pp. 3299–3310.

[30] Jac Fredo Agastinose Ronicko et al. “Diagnostic classification of autism using resting-state fMRI data improves with full correlation functional brain connectivity compared to partial correlation”. In: *Journal of Neuroscience Methods* 345 (2020), p. 108884.

[31] Alexey Natekin and Alois Knoll. “Gradient boosting machines, a tutorial”. In: *Frontiers in neurorobotics* 7 (2013), p. 21.

[32] Abhishek Das, Saumendra Kumar Mohapatra, and Mihir Narayan Mohanty. “Brain Image Classification Using Optimized Extreme Gradient Boosting Ensemble Classifier”. In: *Biologically Inspired Techniques in Many Criteria Decision Making: Proceedings of BITMDM 2021*. Springer, 2022, pp. 221–229.

[33] Lawrence V Fulton et al. “Classification of Alzheimer’s disease with and without imagery using gradient boosted machines and ResNet-50”. In: *Brain sciences* 9.9 (2019), p. 212.

[34] PM Siva Raja and K Ramanan. “Lesion localization and extreme gradient boosting characterization with brain tumor MRI images”. In: *Advances in Data Science and Management: Proceedings of ICDSM 2019*. Springer. 2020, pp. 395–409.

[35] Farzana Z Ali et al. “Gradient boosting decision-tree-based algorithm with neuroimaging for personalized treatment in depression”. In: *Neuroscience informatics* (2022), p. 100110.

[36] Sepp Hochreiter. “The vanishing gradient problem during learning recurrent neural nets and problem solutions”. In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 6.02 (1998), pp. 107–116.

[37] Sepp Hochreiter and Jürgen Schmidhuber. “Long short-term memory”. In: *Neural computation* 9.8 (1997), pp. 1735–1780.

[38] Felix A Gers, Jürgen Schmidhuber, and Fred Cummins. “Learning to forget: Continual prediction with LSTM”. In: *1999 Ninth International Conference on Artificial Neural Networks ICANN 99* (1999).

[39] Felix A. Gers, Nicol N. Schraudolph, and Jürgen Schmidhuber. "Learning Precise Timing with Lstm Recurrent Networks". In: *J. Mach. Learn. Res.* 3.null (Mar. 2003), 115–143. ISSN: 1532-4435. DOI: 10.1162/153244303768966139. URL: https://doi.org/10.1162/153244303768966139.

[40] Hum Nath Bhandari et al. "Predicting stock market index using LSTM". In: *Machine Learning with Applications* 9 (2022), p. 100320.

[41] K. Greff et al. "LSTM: A Search Space Odyssey". In: *IEEE Transactions on Neural Networks and Learning Systems* 28.10 (2017), pp. 2222–2232. DOI: 10.1109/TNNLS.2016.2582924.

[42] Nicha C Dvornek et al. "Identifying autism from resting-state fMRI using long short-term memory networks". In: *International Workshop on Machine Learning in Medical Imaging*. Springer. 2017, pp. 362–370.

[43] Benjamin Lindemann et al. "A survey on long short-term memory networks for time series prediction". In: *Procedia CIRP* 99 (2021), pp. 650–655.

[44] Vikas Khullar et al. "Deep Learning-Based Binary Classification of ADHD Using Resting State MR Images". In: *Augmented Human Research* 6.1 (2021), pp. 1–9.

[45] Rui Liu et al. "Multi-LSTM Networks for Accurate Classification of Attention Deficit Hyperactivity Disorder from Resting-State fMRI Data". In: *2020 2nd International Conference on Industrial Artificial Intelligence (IAI)*. IEEE. 2020, pp. 1–6.

[46] Nicha C Dvornek et al. "Jointly discriminative and generative recurrent neural networks for learning from fMRI". In: *International Workshop on Machine Learning in Medical Imaging*. Springer. 2019, pp. 382–390.

[47] Ramchandra Rimal et al. "Comparative study of various machine learning methods on ASD classification". In: *International Journal of Data Science and Analytics* (2023), pp. 1–15.

[48] Ahmed El-Gazzar et al. "A hybrid 3dcnn and 3dc-lstm based model for 4d spatio-temporal fMRI data: an abide autism classification study". In: *OR 2.0 Context-Aware Operating Theaters and Machine Learning in Clinical Neuroimaging*. Springer, 2019, pp. 95–102.

[49] Kyunghyun Cho et al. "Learning Phrase Representations using RNN Encoder–Decoder for Statistical Machine Translation". In: *EMNLP*. 2014.

[50] Nawa Raj Pokhrel et al. "Predicting nepse index price using deep learning models". In: *Machine Learning with Applications* 9 (2022), p. 100385.

[51] Rahul Dey and Fathi M Salem. "Gate-variants of gated recurrent unit (GRU) neural networks". In: *2017 IEEE 60th international midwest symposium on circuits and systems (MWSCAS)*. IEEE. 2017, pp. 1597–1600.

[52] Kaiming Li et al. "A multimodal MRI dataset of professional chess players". In: *Scientific data* 2.1 (2015), pp. 1–9.

[53] Yan Chao-Gan and Zang Yu-Feng. "DPARSF: a MATLAB toolbox for "pipeline" data analysis of resting-state fMRI". In: *Frontiers in systems neuroscience* 4 (2010).

[54] Hugh A Chipman, Eric D Kolaczyk, and Robert E McCulloch. "Adaptive Bayesian wavelet shrinkage". In: *Journal of the American Statistical Association* 92.440 (1997), pp. 1413–1421.

[55] Stéfan van der Walt et al. "scikit-image: image processing in Python". In: *PeerJ* 2 (June 2014), e453. ISSN: 2167-8359. DOI: 10.7717/peerj.453. URL: https://doi.org/10.7717/peerj.453.

[56] Zini Jian et al. "Research on BOLD-fMRI Data Denoising Based on Bayesian Estimation and Adaptive Wavelet Threshold". In: *Oxidative Medicine and Cellular Longevity* 2021 (2021).

[57] Nilotpal Sanyal and Marco AR Ferreira. "Bayesian wavelet analysis using nonlocal priors with an application to FMRI analysis". In: *Sankhya B* 79.2 (2017), pp. 361–388.

[58] Alle Meije Wink and Jos BTM Roerdink. "Denoising functional MR images: a comparison of wavelet denoising and Gaussian smoothing". In: *IEEE transactions on medical imaging* 23.3 (2004), pp. 374–387.

[59] Karl Pearson. “Correlation coefficient”. In: *Royal Society Proceedings*. Vol. 58. 1895, p. 214.

[60] Stephen M Smith. “The future of FMRI connectivity”. In: *Neuroimage* 62.2 (2012), pp. 1257–1266.

[61] Joanne C. Beer et al. “Incorporating prior information with fused sparse group lasso: Application to prediction of clinical measures from neuroimages”. In: *Biometrics* 75.4 (2019), pp. 1299–1309. DOI: https://doi.org/10.1111/biom.13075. eprint: https://onlinelibrary.wiley.com/doi/pdf/10.1111/biom.13075. URL: https://onlinelibrary.wiley.com/doi/abs/10.1111/biom.13075.

[62] Junghi Kim et al. “Testing group differences in brain functional connectivity: using correlations or partial correlations?” In: *Brain connectivity* 5.4 (2015), pp. 214–231.

[63] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. “Sparse inverse covariance estimation with the graphical lasso”. In: *Biostatistics* 9.3 (2008), pp. 432–441.

[64] R Artusi, P Verderio, and EJTIjobm Marubini. “Bravais-Pearson and Spearman correlation coefficients: meaning, test of hypothesis and confidence interval”. In: *The International journal of biological markers* 17.2 (2002), pp. 148–151.

[65] Jake Lever, Martin Krzywinski, and Naomi Altman. “Points of significance: Principal component analysis”. In: *Nature methods* 14.7 (2017), pp. 641–643.

[66] Naomi Altman and Martin Krzywinski. “The curse (s) of dimensionality”. In: *Nat Methods* 15.6 (2018), pp. 399–400.

[67] I Daubechies et al. “Independent component analysis for brain fMRI does not select for independence”. In: *Proceedings of the National Academy of Sciences* 106.26 (2009), pp. 10415–10422.

[68] F. Pedregosa et al. “Scikit-learn: Machine Learning in Python”. In: *Journal of Machine Learning Research* 12 (2011), pp. 2825–2830.

[69] Stefan Van der Walt et al. “scikit-image: image processing in Python”. In: *PeerJ* 2 (2014), e453.

[70] Martín Abadi et al. *TensorFlow: Large-Scale Machine Learning on Heterogeneous Systems*. Software available from tensorflow.org. 2015. URL: https://www.tensorflow.org/.

[71] Hum Nath Bhandari et al. “LSTM-SDM: An integrated framework of LSTM implementation for sequential data modeling”. In: *Software Impacts* 14 (2022), p. 100396.