Denoising Task-Based Functional Magnetic Resonance Imaging Data Using Hybrid Deep Neural Network Model

by

Sahar Kashfolayat

A Report Submitted in Partial Fulfillment of the

Requirements for the Degree of

Master of Engineering

In the department of Electrical and Computer Engineering

University of Victoria

University of Victoria

Supervisory Committee
Denoising Task-Based fMRI Data Using Hybrid Deep Neural Network Model

by

Sahar Kashfolayat

University of Victoria, 2022

Supervisory Committee

Dr. Amirali Baniasadi, Department of Electrical and Computer Engineering

Supervisor

Dr. Mihai Sima, Department of Electrical and Computer Engineering

Departmental Member

Table of Contents

List of Figures

Acronyms

BOLD:	Blood Oxygen Level
DNN:	Deep Neural Network
fMRI:	Functional Magnetic Resonance Imaging
GLM:	Generalized Linear Model
GM:	Gray Matter
LSTM:	Long Short-Term Memory
MRI:	Magnetic Resonance Imaging
NMR:	Nuclear Magnetic Resonance
RNN:	Recurrent Neural Network

Acknowledgments

I would like to express my gratitude to my supervisor, Dr. Amirali Baniasadi, for his constant guidance, support and wisdom throughout my research and my program.

Dedication

To my grandmother and my father in heaven who hold a very special place in my heart. To My mother, her unconditional love and support throughout all stages of my life has known no bounds. Their encouragements all through my life are a great reason for my success. I dedicate this to all three of them.

Abstract

Error management is one of the most important concerns in medical science. A doctor's ability to make an accurate diagnosis can save a patient's life. Therefore, it is essential that doctors have access to accurate medical data in order to make the best possible decisions for their patients. This means that technology can help doctors be more accurate in their diagnoses and reduce the number of mistakes they make. In this study, a hybrid deep neural network (HDNN) is proposed to reduce the noise in task-based functional magnetic resonance imaging (fMRI) data. This model has been developed to improve the performance of the Deep Neural Network (DNN) model [1]. This network, like the DNN model, has a sequential structure and utilizes the same cost function and the same algorithm to update network parameters. The architecture of the model consists of three common layers and four non-common layers to perform noise reduction. The first three layers consist of two one-dimensional convolutional filters to reduce physiological noise in both GM and non-GM data and one LSTM layer to consider the temporal correlation of input data [9]. The final layers are used to separate and select data respectively, including a fully connected and a conventional selective layer for GM and non-GM input data (with different resolutions) which introduces the hybrid operation of the network based on the type of input data. In order to analyze the effect of these modifications on the output image, using the same conditions as described in [1], simulated and real fMRI data are used to compare DNN and HDNN models. The results show that the proposed model was able to reduce noise better than the DNN model by at least 5% in GM areas and 83% in non-GM areas.

Chapter 1: Introduction

1.1 Context

Imaging technologies help us to discover and illustrate the structure and function of the brain, leading to advancements in brain activity research and studies. MRI is a cross-sectional imaging technique that produces images of the physical or chemical properties of an object from its NMR signal [2]. This imaging can be processed in a resting condition or while doing an activity. The individual completes a specific task in the activity-based scenario, and then the observed signal coupled with resting-state data is used to determine the activity pattern of a brain area related to the task. Blood oxygen level dependence (BOLD) fMRI, which is based on variations in blood oxygen due to brain activity, is one of the most prevalent approaches in functional imaging of the brain. This method, a non-invasive method that started in 1991, is used as the dominant tool for brain imaging and has the ability to map any activity to its origin [3], [4]. Activity in one part of the brain requires more oxygen to be delivered to that area, and so physiological events such as increased cerebral blood flow, increased blood volume, and oxygen metabolism accompany it. Based on this, it is possible to have a dynamic map of brain activity by considering the oxygen level of venous blood using MRI imaging. Principles of BOLD functional imaging that changes in the concentration of deoxyhemoglobin in a specific area of the brain can alter the intensity of the MR signal received from that tissue. The BOLD response can be assumed as shown in Figure 1.1. With the onset of stimulation, a signal reduction of 0.5 to 1 second is observed in this diagram, which is caused by a delay in blood supply and is essentially a sign of signal reduction due to the consumption of oxygen in the tissue before oxygenation. After that, the BOLD response increases for 8 to 20 seconds and returns to baseline with a delay when the stimulus stops. It should be noted that the intensity of the magnetic field is very effective in the rate of change of this signal and higher intensity can make more differentiation in the signal [5], [6].

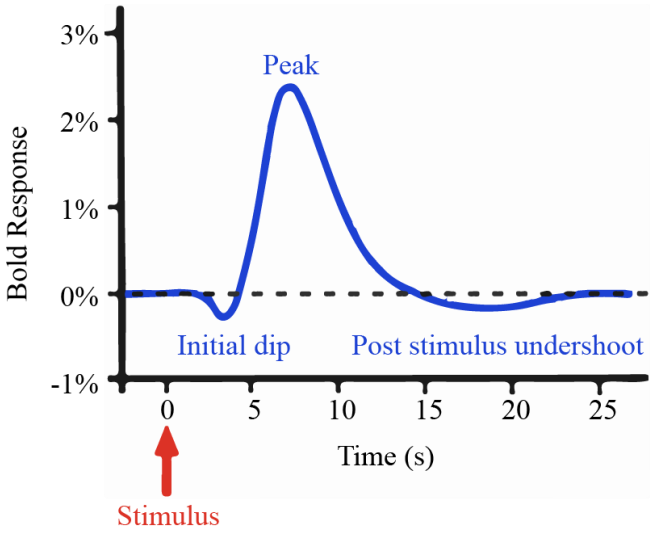


Fig. 1.1 Example of BOLD response signal related to stimulation of subject over time

1.2 Project Objectives

The proposed model uses a similar structure and parameters for network operation utilized in DNN research [1] and attempts to reduce the output image noise through deep neural network redesigning (figure 1.2). The DNN model processes GM and non-GM fMRI images using the same structure. The goal of this study is to create a hybrid mechanism in the neural network structure that allows the so-called components to be processed differently, lowering the cost function and leading to better results.

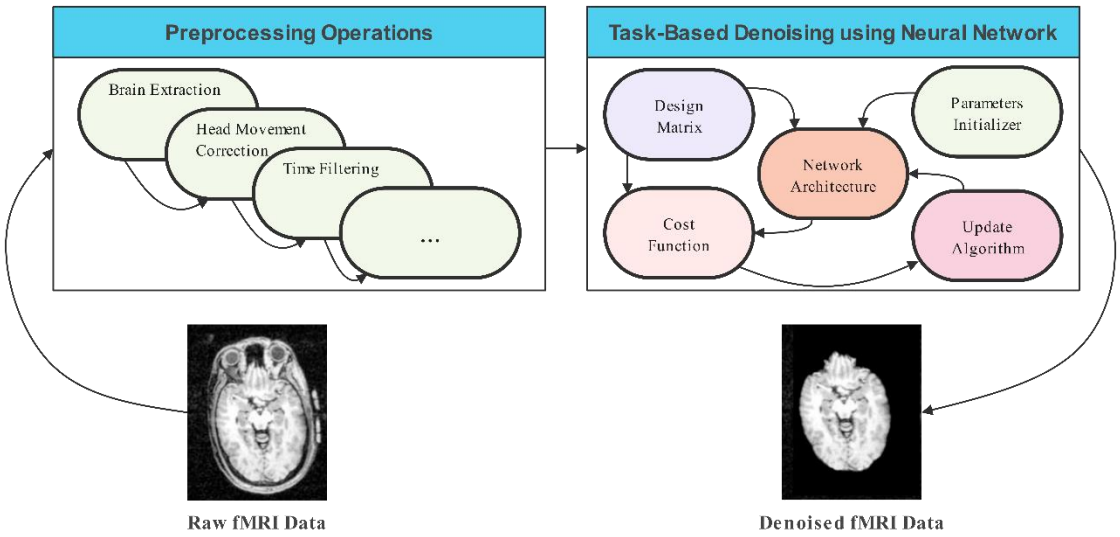


Fig. 1.2 Demonstration of operational procedures followed in the proposed study to denoised fMRI data

1.3 Report Outline

The remaining chapters in this report are structured as follows. Chapter 2 gives an overview of existing DNN models and gives a summary of key features of the DNN model and goes on with the cost function. Chapter 3 outlines the proposed model to address better cost function. Chapter 4 depicts the experimental evaluation of the proposed scheme. Chapter 5 makes concluding remarks.

Chapter 2: Background

2.1 Preprocessing

Working with neural data presents its own set of challenges, thus concentrating on all the steps that will make subsequent processing easier should not be overlooked. The preprocessing technique encompasses all of the procedures that allow raw data to be evaluated. So, preprocessing is one of the vital actions which will be done in different steps which will be explained below.

2.1.1 Brain Extraction

We separate non-brain tissues like the skull and scalp from the brain at this point. This is normally accomplished by comparing changes in image intensity. The brain extraction procedure would also include T1 and T2 pictures. The photos are prepared for adaption and segmentation in this step. The brain extraction is implemented through SPM software [7] and an example is shown in figure 2.1.1.

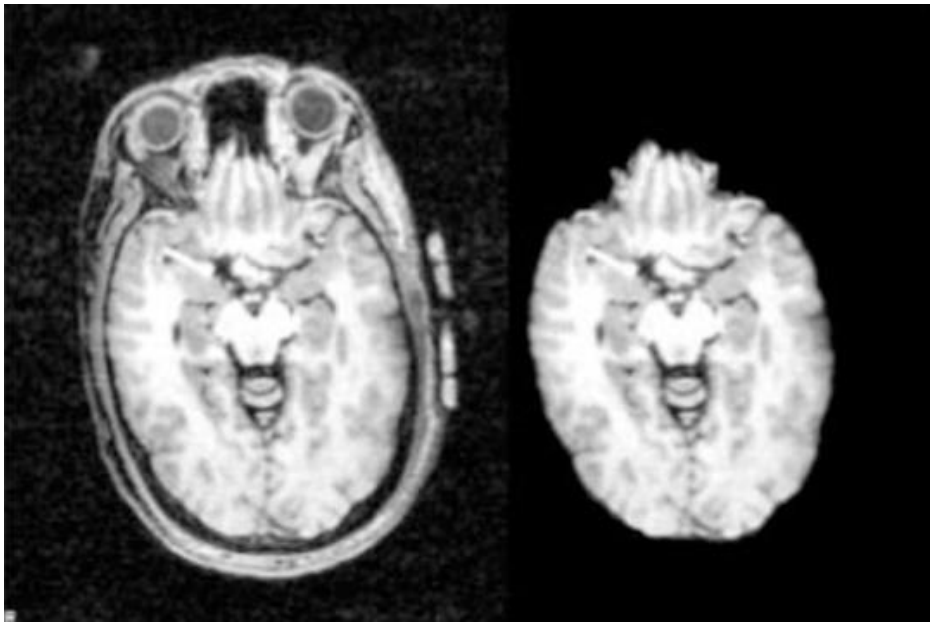


Fig. 2.1.1 Image with the skull and scalp (left) is processed to extract image of the brain (right).

2.1.2 Head Movement Correction

Head movement correction is a critical preprocessing in fMRI to ensure the accuracy of subsequent analytical processes due to the unusually long image capture period as shown in figure 2.1.2. The person's head may move very little during the scans, which can alter the captured images. As a result, small head movements must be corrected by adapting images taken at different times to a reference image (usually the average image) so that the images of the different time frames match exactly spatially. This is done to ensure that a fixed area of the brain is in all images and that the subsequent processing and analysis stages are accurate. In any event, a complete image of head movement is not achievable, and the consequences of movement inside the slide screen that leads to incomplete stimulation of a slice will not be compensated across many scenarios [8].

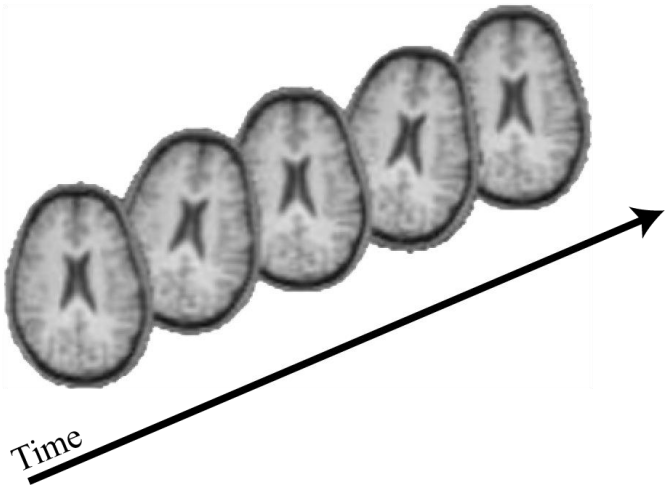


Fig 2.1.2 Effect of head movements in captured images over time

2.1.3 Time Filtering

The most common use of time filtering is to remove undesirable frequency components from images. Noise from physiological parameters such as respiration and heart rate, as well as thermal noise from the scanner, are among these components. The power source also has an impact on the noise level. Time filtering should not be so strong that it degrades the activity signal. [8].

2.1.4 Spatial Smoothing

Spatial smoothing is commonly used to improve the image's SNR. Since this is commonly accomplished using Gaussian smoothing kernel, the filter dimensions should roughly correspond to the spatial dimensions of the active area [8]. The degree of image smoothing can have an impact on the final output. The smoothing effect on activity detection is demonstrated in Figure 2.1.4.

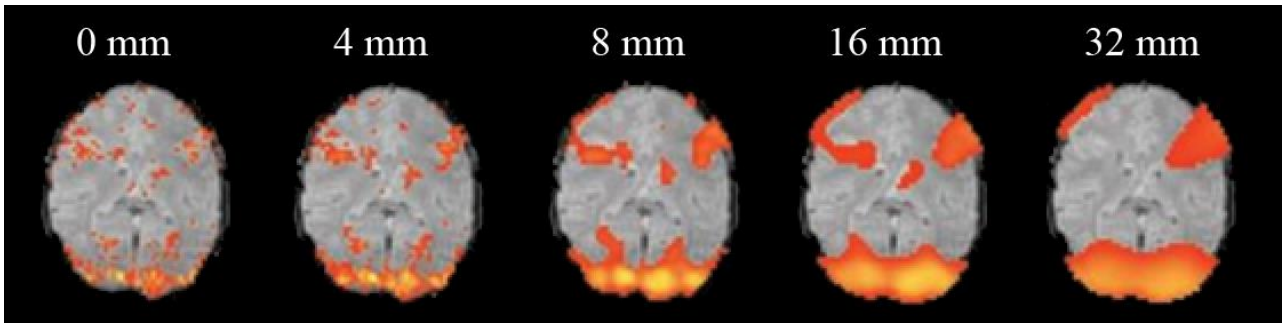


Fig 2.1.2 An example of the effects of spatial smoothing on activity, the numbers show the size of the Gaussian smoothing kernel applied to the data before analysis. Increased smoothing detects larger clusters but reduces the detection of smaller clusters.

2.1.5 Slice Timing Correction

Each slice of fMRI is collected at a distinct time. In fMRI analysis, however, it is assumed that all of them associated with a volume are captured at the same time. At this point, our goal is to compensate for this inequality. This correction is much more critical for the evaluation of resting-state fMRI data, and improves the final sensitivity of fMRI data dependent on a given task [8].

2.1.6 Image Matching

Since images should be captured at a quick rate, typically they have a low spatial resolution. Image registration using high-resolution anatomical images is frequently utilized to display the active areas of the brain accurately. This method converts functional images to the anatomical image space of a subject, using transformations with 6 to 12 degrees of freedom. Non-linear transformations can provide greater accuracy. Image matching is an intermediate step before taking the functional image of a subject to the standard atlas space, and increases the accuracy of the final conversion [8].

2.1.7 Match the Images to the Standard Atlas

In the previous stage, we demonstrated how to adjust the functional image of a subject to the structural image of that subject. After that, the images should be matched to the standard atlas so that the findings of this study may be compared to those of other studies using different procedures and subjects. The Talairach Atlas [9], derived from MR images of a 60-year-old right-handed European female, is a three-dimensional coordinate system of the human brain that is used to map the location of brain structures regardless of individual variances in brain size and form. It is created using a huge set of MRI scans of normal people in a control group and is one of the most widely used brain atlases [9]. The following figure demonstrates the procedure to match a functional image to the structural one.

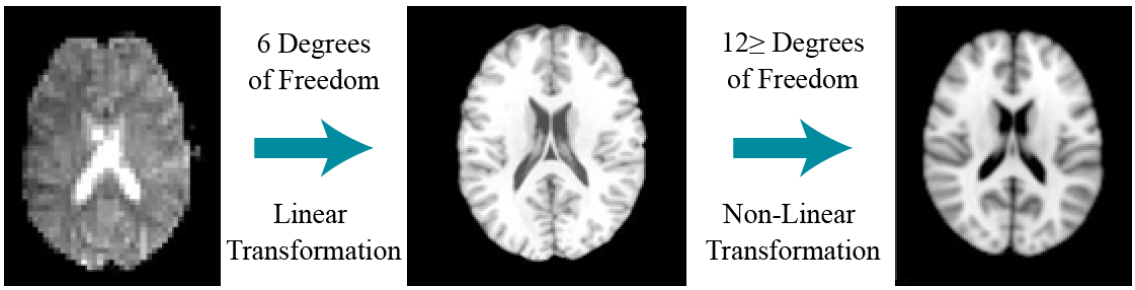


Fig 2.1.7 two stages of matching functional image with structural image: 1) linear method and 6 degrees of freedom and 2) nonlinear method with 12 degrees of freedom or more adapting to the standard atlas.

2.2 Statistical Modeling

The purpose of fMRI data analysis is to see if changes in the BOLD signals, related to the time series of each voxel, are in response to stimulation. For example, if a stimulus in a block pattern such as the red line shown in Figure 2.2.a is applied to the subject on a regular basis, we are looking to extract a time series for each voxel that resembles the BOLD signal (the blue line). The tool used to fit and detect these changes is the generalized linear model (GLM) in which the following equation is assumed as the basis [9].

$$\boldsymbol{Y} = \boldsymbol{X\beta} + \boldsymbol{e} \qquad \text{(Eq. 2.2.a)}$$

In this formula, Y represents the voxel time series, X represents the design matrix, which is determined according to the experimental paradigm, β is the unknown model parameters and e represents the modeling error [9].

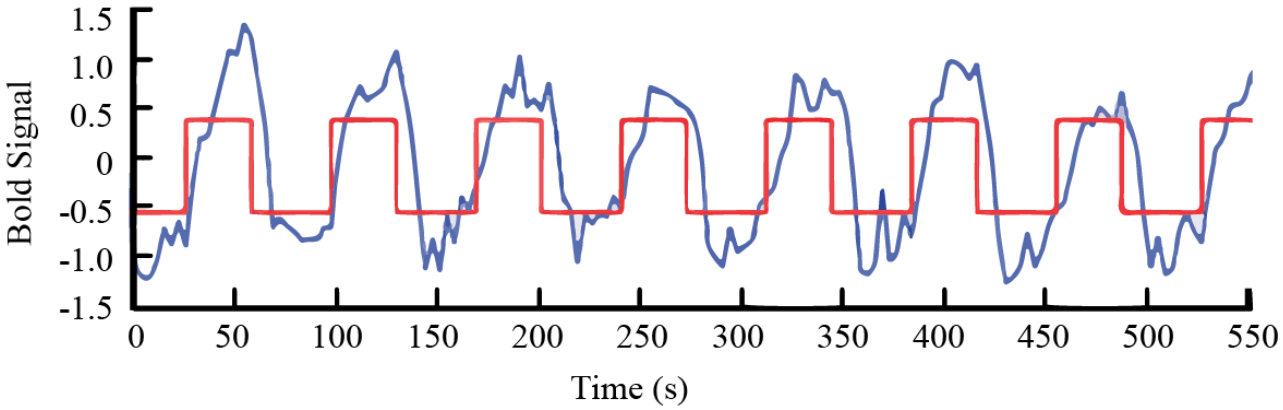


Fig. 2.2.a The BOLD fMRI time series in an active voxel: The BOLD signal is shown for an active voxel (blue) and an excitation time series (red)

A fundamental parameter in this statistical model is the design matrix that is needed to run the simulation procedure since it is a vital part of the selection layer in neural network architecture which is explained thoroughly in the next section. The design matrix is obtained by convolving binary tasks, which are available in the imaging information database, with the hemodynamic response function (HRF) which is shown in figure 2.2.b. Binary tasks describe time frames in which a stimulus is applied to a subject. Since HRF is the BOLD signal generated by an impulse stimulation, that peaks in approximately 6 seconds and then returns to its previous state after a few seconds [1], if we convolve the binary task with HRF, we can represent a signal that simulates the BOLD response related to stimulation pattern (binary tasks) during fMRI scans. Therefore, if the processed fMRI image is compared to this generated design matrix, more similarity should mean more signal level generated by blood oxygen level in the image and less presence of noise.

The hemodynamic response function is modeled in two ways. If we simply assume a Gamma distribution for this function, the modeled HRF is called the canonical HRF. On the other hand, to assume a model with more flexibility, we can use the summation of weighted basis functions [12], to describe HRF mathematically.

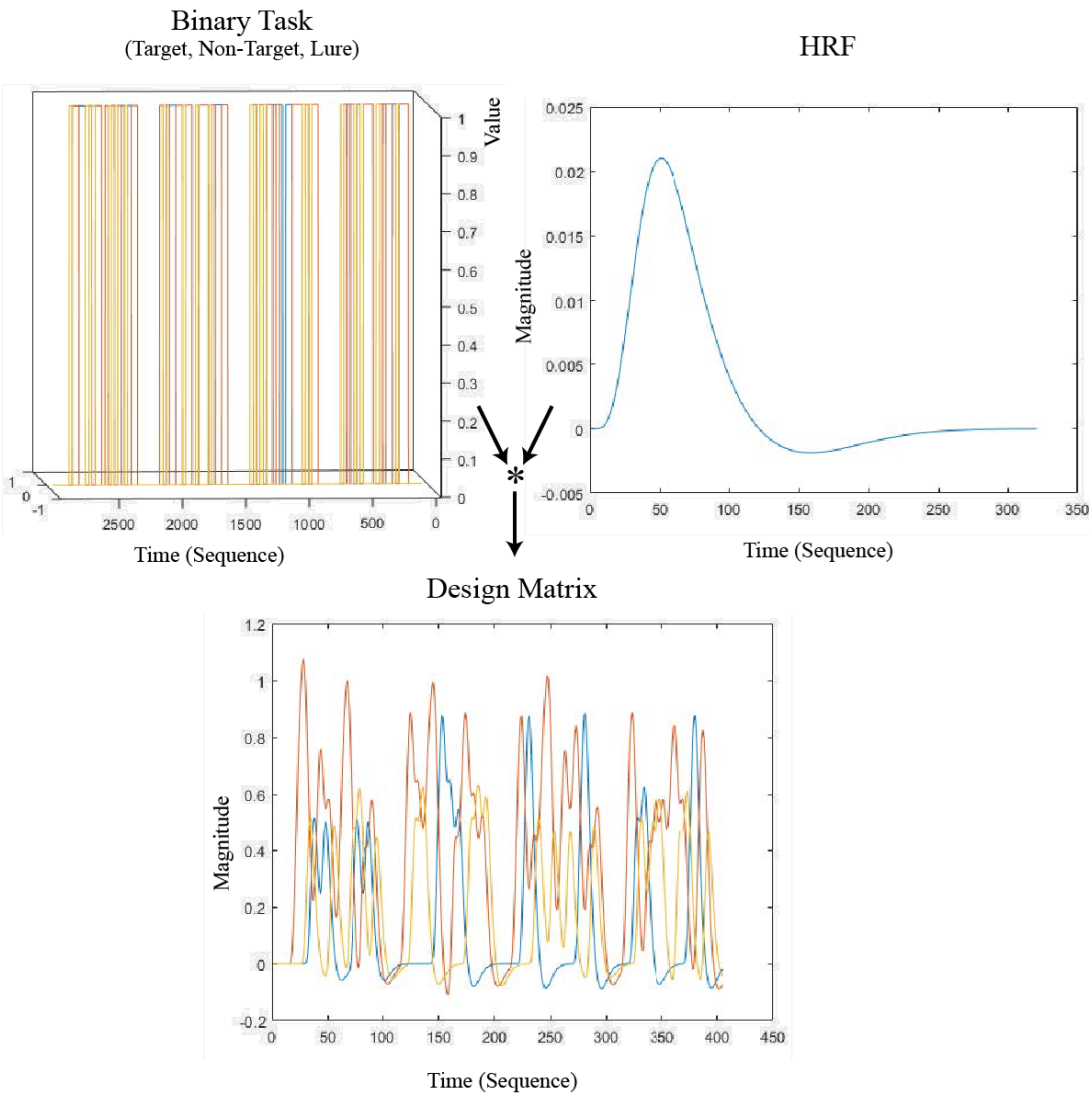


Fig. 2.2.b General illustration of Design Matrix calculation

2.3 DNN Model

The model used in DNN research [1] is based on a deep neural network structure along with the definition of a custom cost function that maximizes the correlation between GM fMRI data and the design matrix which leads to less noise in task-based fMRI data.

2.3.1 Architecture of Four-Layer DNN Model

The neural network proposed in DNN model includes four general layers: A 1-dimensional convolutional layer (1-D Convolution), one LSTM layer, a time-distributed fully connected layer, and a selection layer. Figure 2.3.1.a shows the structure of this model. DNN model receives data related to a voxel over time as input. To avoid unsaturated signals, the first 15 volumes of fMRI data are discarded. As the model describes, to feed 4-Dimentional fMRI data to the network, we need to convert the 3-dimentional data over time to 1-dimentional data in time frames [1].

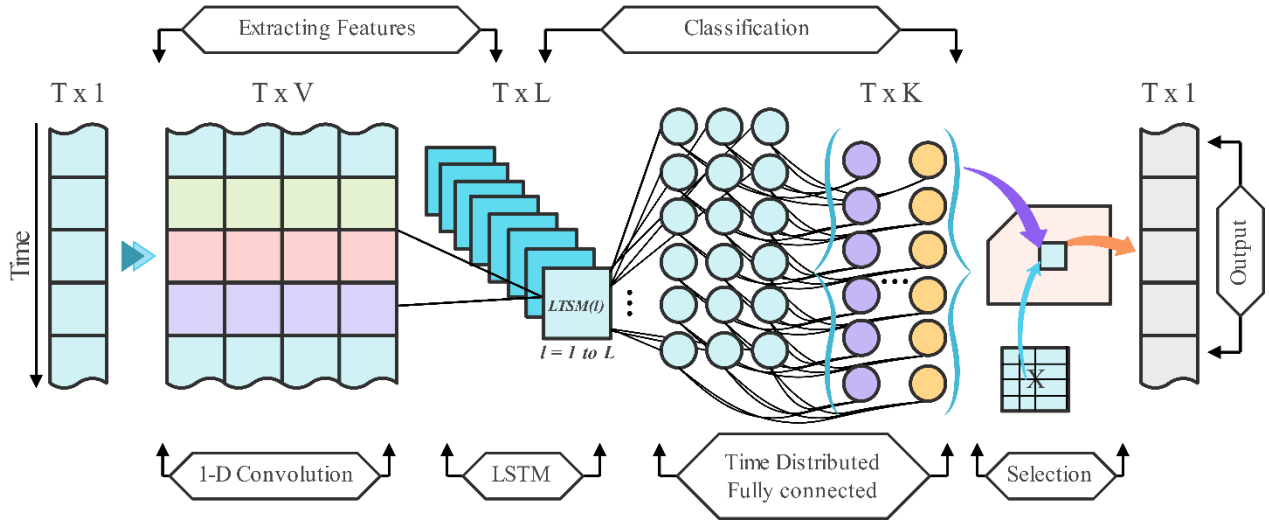


Fig. 2.3.1.a Structure of layers in DNN model

1-D convolutional layer employs temporal filters that unlike general designs does not have a pre-defined frequency threshold. The noise cancellation procedure is performed independently for each data point because the model is data-driven. Figure 2.3.1.b represents a simple one-dimensional convolutional filter with filter size 5 and stride length 1. To consider correlation of data between time frames for each voxel, a subset of RNNs are used. LSTM networks are a common type of RNN networks that are more flexible [1].

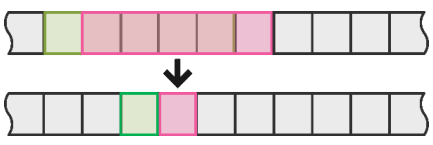


Fig. 2.3.1.b Depiction of 1-D convolutional Layer

The third layer provides another means for the classification of the processed data. Since we need to keep the temporal state of data at the output layer, this layer utilizes time-distributed fully connected nodes. The fully connected network and time distributed fully connected network are illustrated in figure 2.3.1.c. It is clear that in this model we need to keep the dimension of data related to time so the fully connected layer is not a suitable classifier.

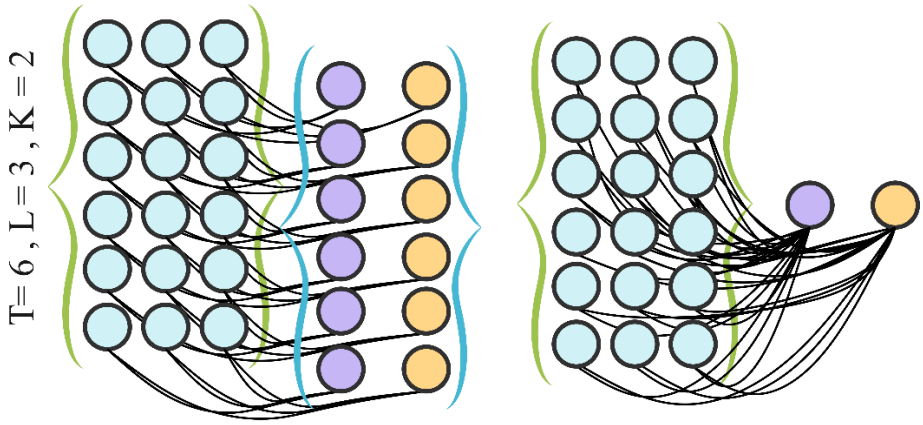


Fig. 2.3.1.c Time distributed fully connected (left) and fully connected (right) networks

The final layer is a selective operational one that provides the output (same size as the input vector) based on the maximum correlation between the time series outputs of the time distributed fully connected layer and the design matrix (X). More details about the functionality of this architecture are provided in the following sections [1].

2.3.2 Definition of Cost Function

The cost function is a criterion for updating neural network parameters. Though there are many cost function definitions available like mean squared error, mean absolute percentage error, cross-entropy and etc. which are generally used in machine learning algorithms, most of these functions can only be used for classification or regression in supervised learning scenarios. Since we do not have access to the perfect class in order to train the network, supervised training is not possible and therefore we cannot employ the former cost functions. Therefore, the DNN network develops its own optimal cost (loss) function as described in eq. 2.3.2 [1]:

$$\mathrm{L}\left(Y_{denoise}, \tilde{Y}_{denoise}, X\right) = -\left(\sum_{y \in Y_{denoise}} r(y, X) - \sum_{\tilde{y} \in \tilde{Y}_{denoise}} r(\tilde{y}, X)\right) \qquad \text{Eq. 2.3.2}$$

Since non-GM voxels are expected to have no neural response, this loss function is minimized when the correlation (Pearson correlation coefficients) between GM denoised fMRI data ($y \in Y_{denoise}$) and the design matrix (X) is maximized and non-GM denoised fMRI data ($\tilde{y} \in \tilde{Y}_{denoise}$) has the minimum correlation with the design matrix. As it is shown in eq. 2.3.2, the optimization of network parameters based on this cost function is only possible following the availability of these factors in each training sequence. The Pearson correlation coefficient is calculated based on the general linear model (GLM). For the purpose of validation of the raining procedure, 90% of GM data are selected randomly and are fed to the network. Random non-GM voxels are also selected to construct a pair of input data with GM ones for each training stage. The remaining 10% GM data along with the random non-GM data (the same batch size) are used as a validation set to monitor whether the network is over-fitted or under-fitted. It should be noted that the initial values of the network parameters are derived from Xavier uniform initialize Adam stochastic gradient-based optimization algorithm is used to update these parameters based on the value of cost function in each iteration [1].

2.4 HCP Database and Raw Material

The MRI data used in this study were obtained from the Human Connectome Project (HCP) database [13]. Structural data of subjects including resting-state MRI and task-based fMRI data related to 7 different tasks are available in this database for 88 subjects. DNN model focuses on the working memory task. Therefore, only resting-state MRI and working memory fMRI data will be used for further analysis [1]. It should be noted that all data available in the HCP database are minimally pre-processed [13].

Structural data masks for areas of interest are used to extract GM and non-GM data from the fMRI dataset. Because there is no neural activity in non-GM parts of the brain (WM and cerebrospinal fluid), the detected signal should be considered as noise and may aid in Denoising the BOLD signal related to neural activity in GM areas. Therefore, a non-GM mask will be used to extract these parts of the data. To reduce minor volume effects, cerebrospinal fluid data is eroded once and white matter data is eroded up to four times, provided that the white matter mask has at least 10,000 voxels after erosion.

2.5 Simulation of Model

The simulated data that are used in this study to investigate the performance of DNN model, are prepared according to the details in [1]. The DNN model is trained for 20 iterations using two fixed datasets. The main purpose of the model is to decrease the non-GM correlation and increase the GM correlation with the design matrix (X), which results in the difference between the two correlations being considered as an error (cost) and be used to optimize the model parameters of the network.

There is no way to calculate a fixed threshold for BOLD signal analysis in order to categorize voxels as active or inactive. However, to reduce the processing load of simulations, considering that neural activity mostly occurs in GM voxels, we calculate the average value of correlation between GM voxels' data and the design matrix. GM voxels that have a correlation value greater than this average value are assumed to be active. Since we need a pair of GM and non-GM data for model analysis, the same number of non-GM voxels are selected randomly and is fed to the network. This procedure is shown in figure 2.4.a.

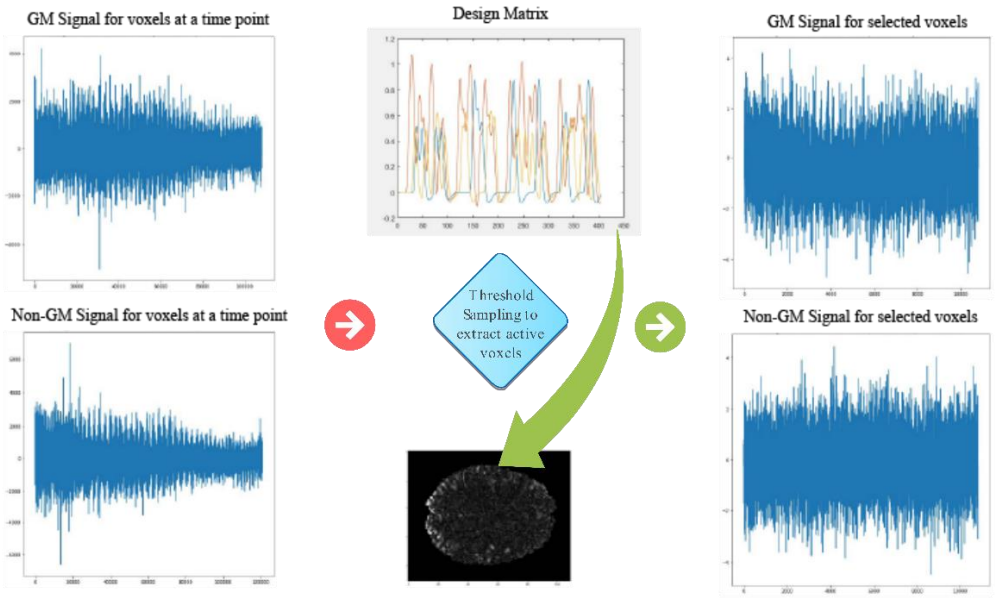
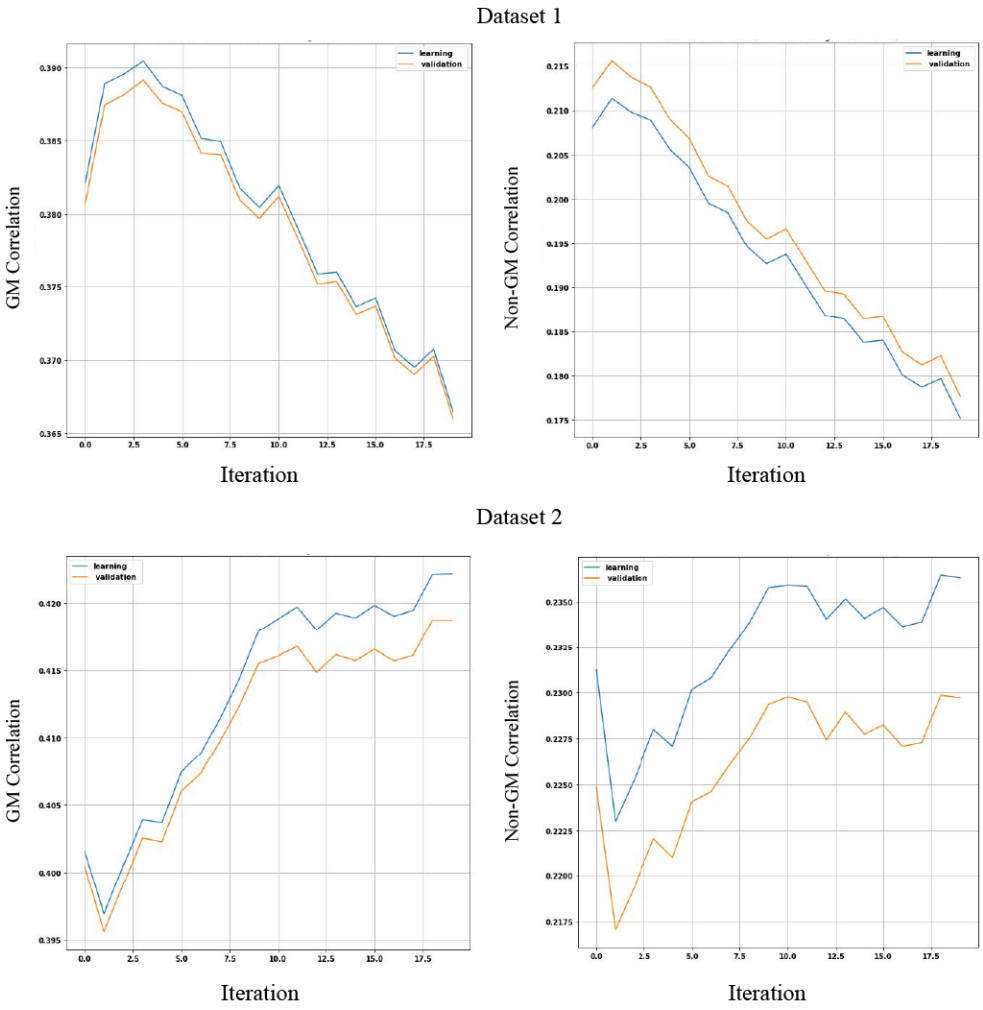


Fig. 2.4.a Data sampling based on the active voxel definition

Figure 2.4.b shows the average correlation values of GM and non-GM simulated data with the design matrix at the output level. The difference between these averages represents a negative scaled value of the defined cost function (eq. 2.3.2), therefore the increment in it after each iteration approves the

reduction of the cost function and successful operation of the learning algorithm. It should be noted that the little difference between validation and training curves suggests that the network is not under-fitted or over-fitted for these sets of data.



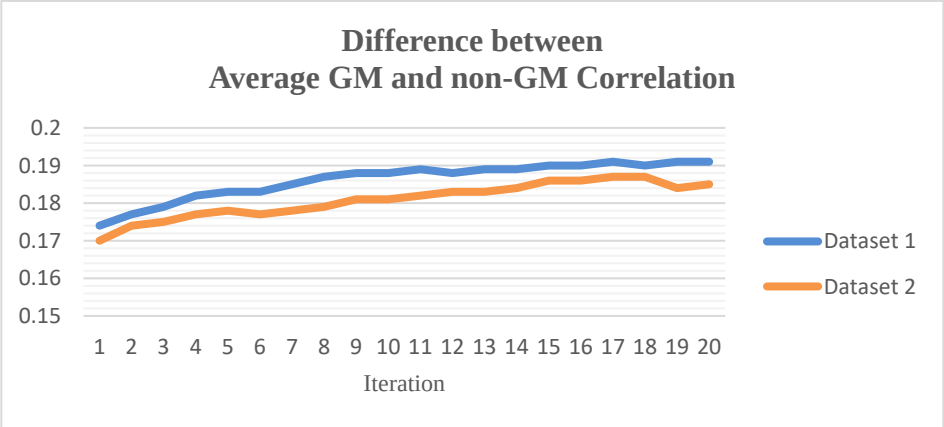


Fig. 2.4.b GM and non-GM correlation values for two sets of data in the DNN model simulation

Chapter 3: Proposed Model

3.1 An overview to Multi-Layer HDNN Model

The Hybrid DNN model (HDNN) has been developed to improve the performance of the DNN model which was explained in previous chapter. In this network, same as the DNN model, a sequential structure is utilized. The model includes of three common layers and acts in a conditional way in the final layers. The first three layers consist of two 1-dimensional convolutional layers (with filter dimensions of [8, 30] and [4,5] and movement step equal to 1) to reduce physiological noise for both GM and non-GM data and one LSTM layer to maintain the temporal correlation of inputs. The final layers are used to select and separate the results, which includes a fully connected layer and a selection layer. The conditional action of this model is based on the type of data that is fed to network. If GM data is used as a time series at input level, a time distributed fully connected layer with four nodes as output will be used. On the other hand, to achieve less correlation between non-GM processed data and design matrix and consequently minimizing the cost function, the resolution of Time distribute fully connected layer for non-GM data is reduced. We employ a time distributed fully connected layer with only two output nodes for non-GM inputs. The following figure illustrates the architecture of the HDNN model.

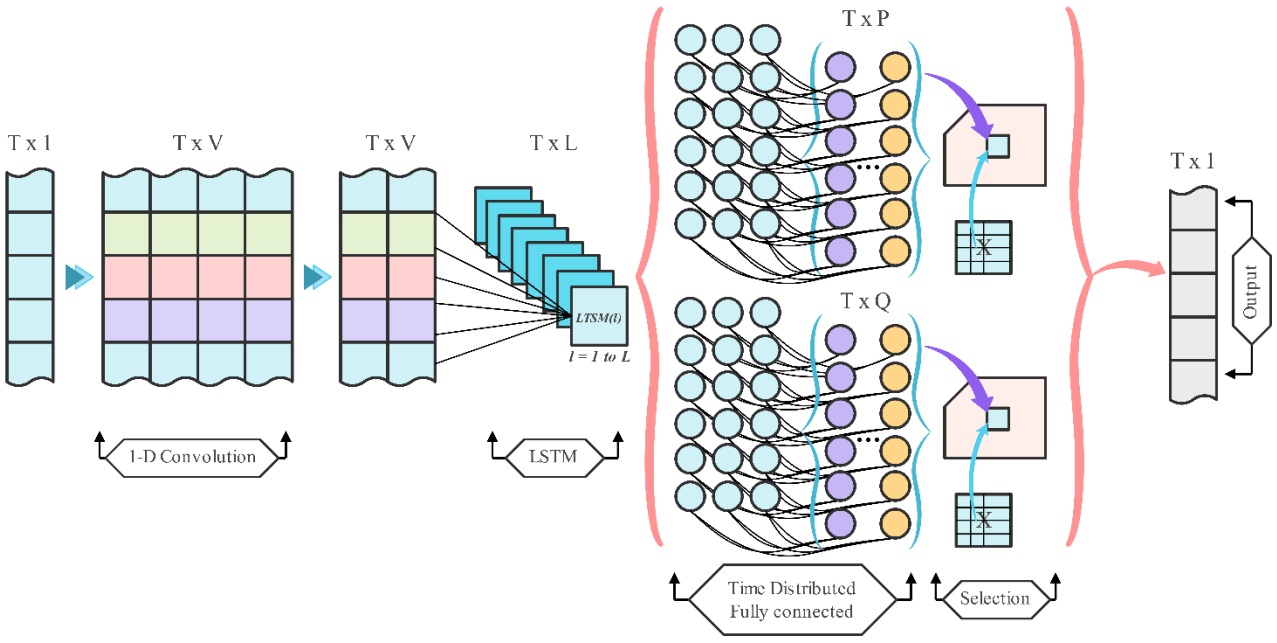


Fig. 3.1 The architecture of the Hybrid DNN model. Conditional operation can be observed in the final stages of data processing

To compare the effect of this modification with the original DNN model on Denoising procedure, the same cost function, initialization and optimization algorithm is used. Also, 90% of the GM and non-GM data (voxels) are randomly selected to train the network and the remaining 10% of data (the GM voxels and random non-GM voxels which are not used in training stage) are used to monitor the state of network whether it is over-fitted or under-fitted.

3.2 Model Simulation and Performance Results

The same simulation process is used to evaluate the HDNN model. Again, two sets of fixed data are used to generate average GM and non-GM correlation values for both training and validation stages in 20 iterations. The results are shown in figure 3.2.a. The difference between average correlation values, again, is a negative scaled cost function (eq. 2.3.2) and shows that after each training iteration the cost function is successfully reduced. Comparing these results with figure 2.4.b, we can see that this difference has a greater average value therefore the cost function is much more reduced, since HDNN is able to reduce non-GM correlation significantly as well as increase GM correlation with the design matrix. Again, it is obvious that the training procedure does not suffer from under-fitting or over-fitting for these sets of data. It is clear that the model performed well for the two simulated data sets and compared to the DNN model, HDNN was able to achieve at least a 5% increment in average GM correlation and an 83% reduction in average non-GM correlation with the design matrix. The average correlation of GM data in six active regions of the brain is shown in figure 3.2.b which shows a minimum average increment of 4.7% in the HDNN model compared to the DNN model. Though the proposed model was not able to perform better than the DNN model in region 4 using dataset2, generally it produced better results. Figure 3.2.c represents the activity mask created for the two simulated data sets, considering the definition of an active voxel.

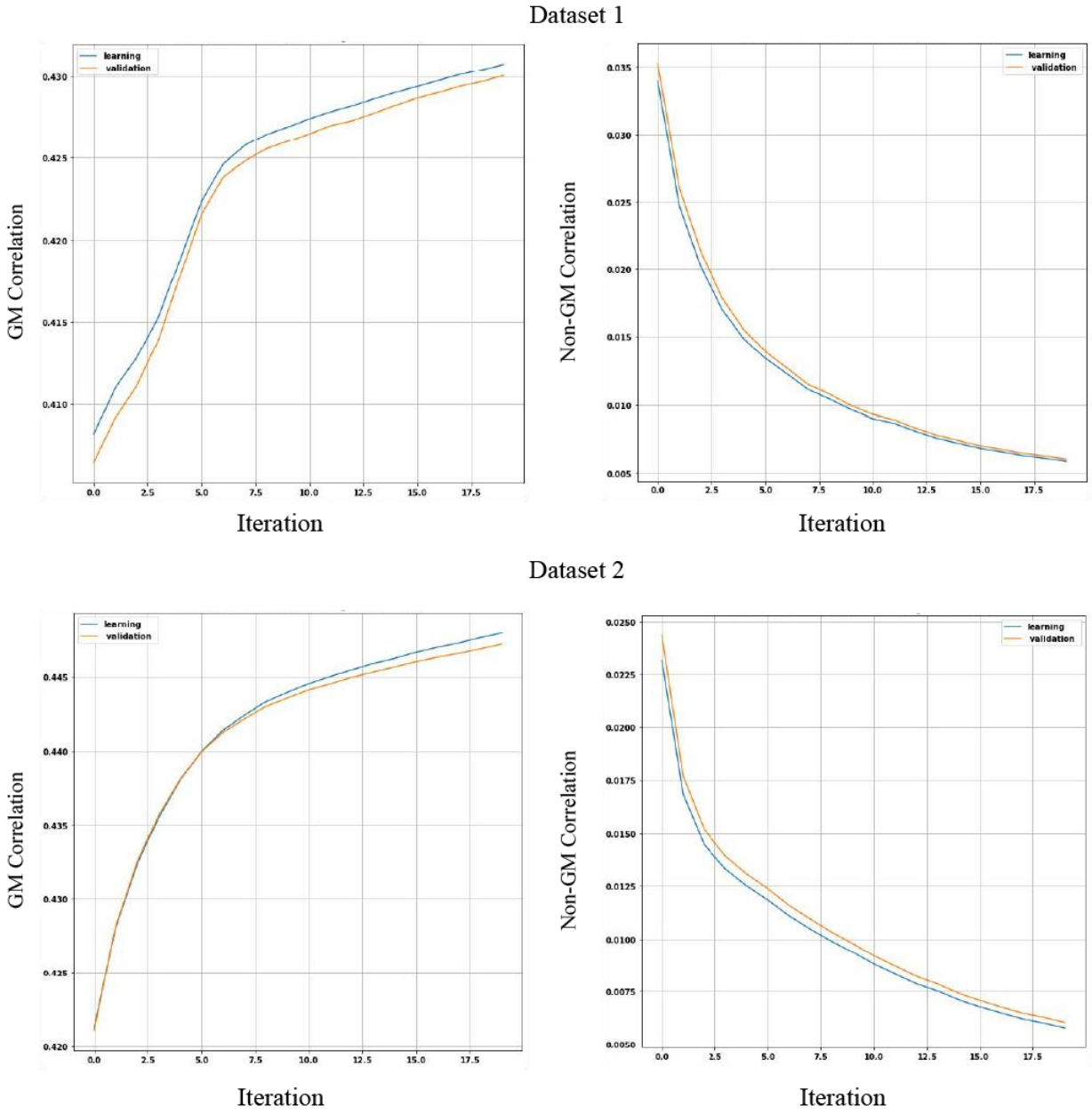
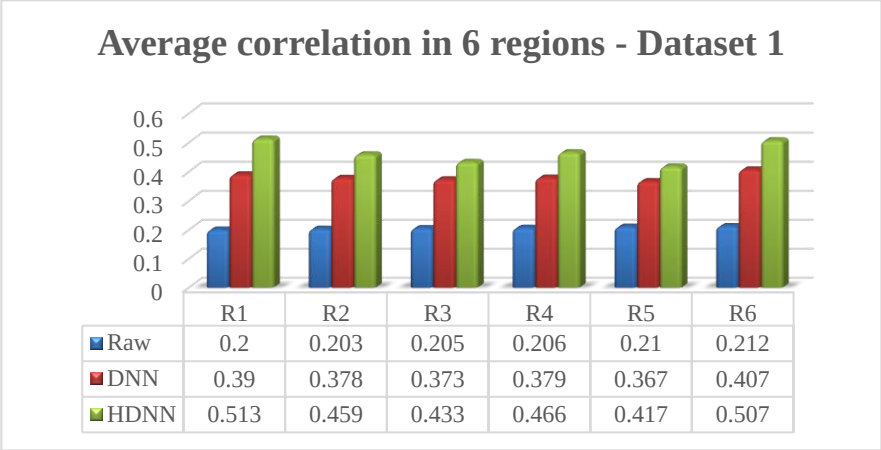
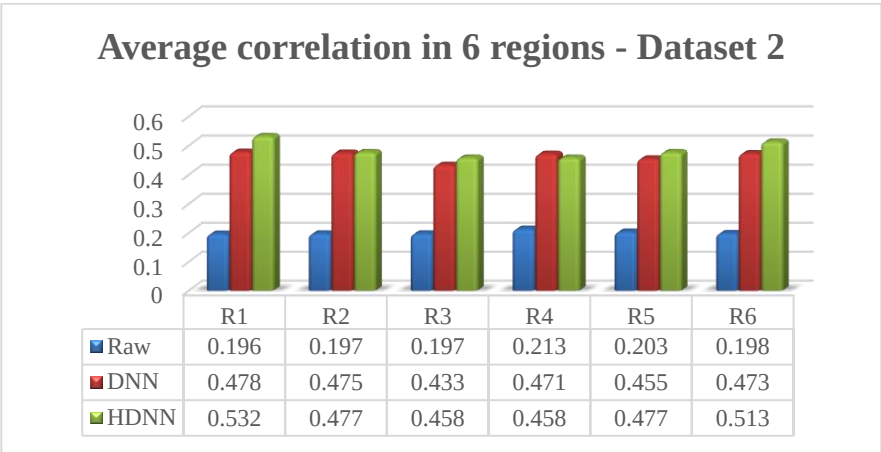


Fig. 3.2.a GM and non-GM correlation values for two sets of data in the HDNN model simulation



	R1	R2	R3	R4	R5	R6
Raw	0.2	0.203	0.205	0.206	0.21	0.212
DNN	0.39	0.378	0.373	0.379	0.367	0.407
HDNN	0.513	0.459	0.433	0.466	0.417	0.507



	R1	R2	R3	R4	R5	R6
Raw	0.196	0.197	0.197	0.213	0.203	0.198
DNN	0.478	0.475	0.433	0.471	0.455	0.473
HDNN	0.532	0.477	0.458	0.458	0.477	0.513

Fig. 3.2.b Numerical results of simulation for two datasets

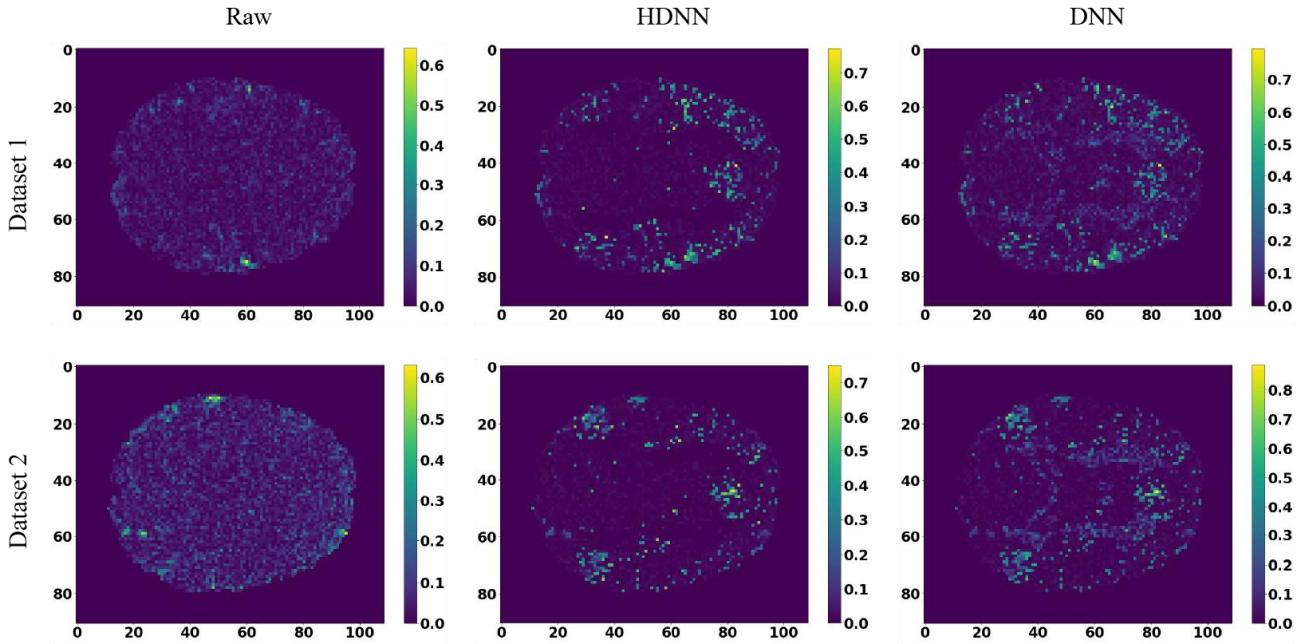


Fig. 3.2.c Activity mask for two sets of simulation data: derived from 1) Raw data 2) DNN model 3) HDNN model

3.3 Performance Analysis of the HDNN Model

3.3.1 Simulated Data Analysis

The results of HDNN model for two selected samples was discussed in the previous sections. We are going to analyze different simulated data sets and real data sets to evaluate the performance of the proposed model. In this research, three simulated data have been generated, which two fixed samples are already used to generate proper outputs. Therefore, the final simulated data is fed to the model and comparison of results are available in Figure 3.3.1.a.

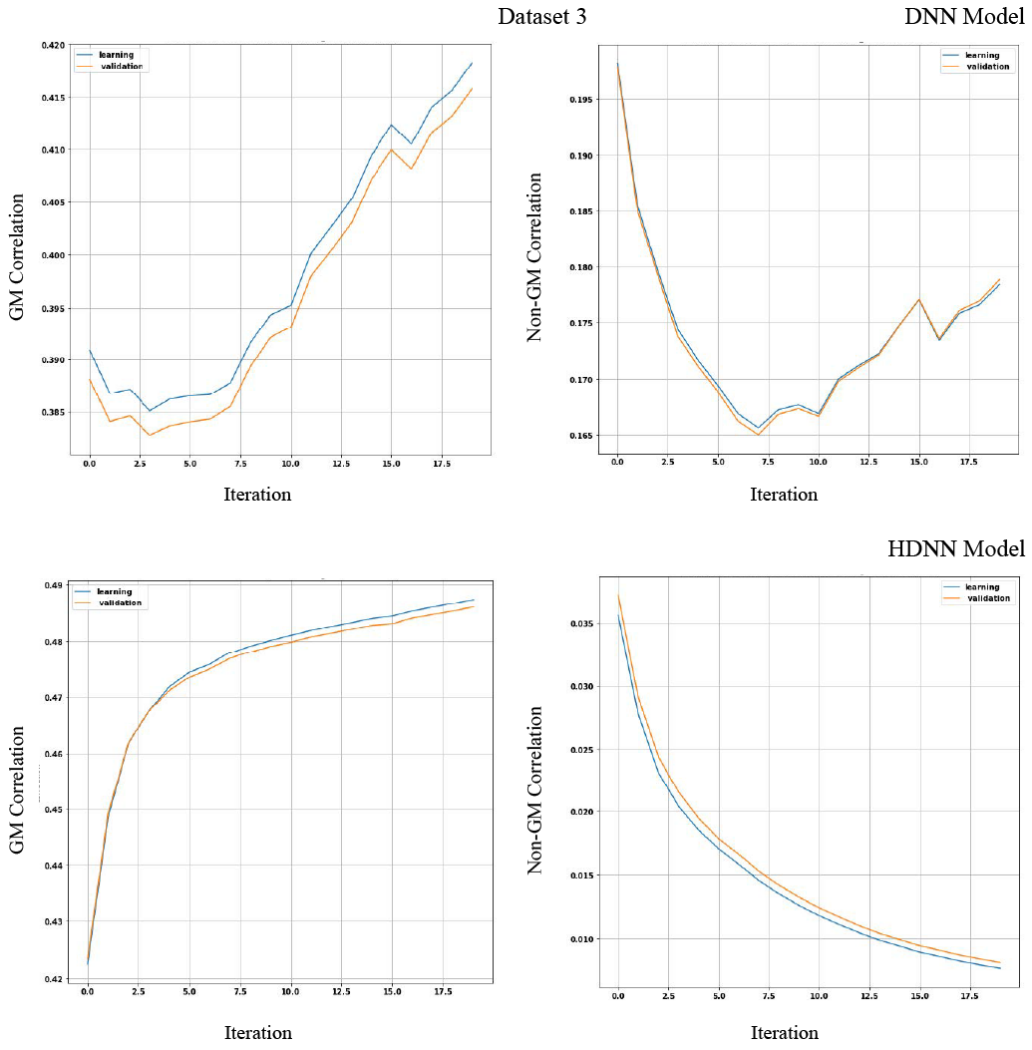
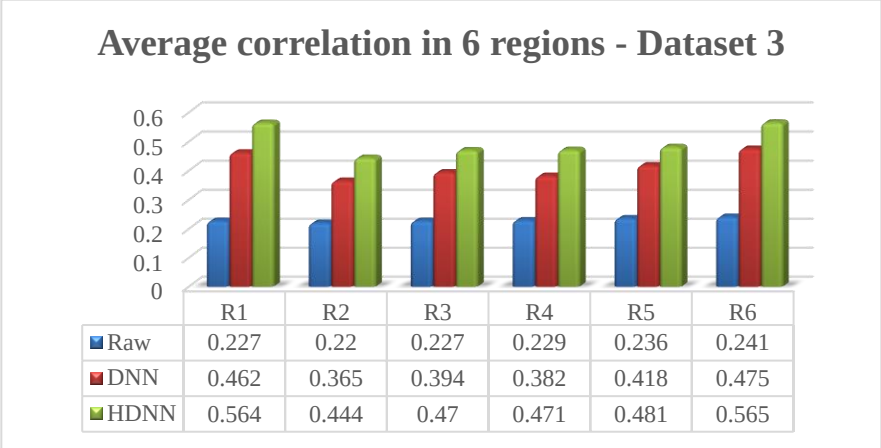


Fig. 3.3.1.a GM and non-GM correlation values for third set of data using DNN and HDNN models

The results show that HDNN model, again, has smoother changes in correlation compared to the DNN model. Average correlation values in six regions for both models and raw data are shown in figure 3.3.1.b. According to the mean correlation changes in 6 regions, it can be seen that the proposed model was able to increase the correlation in GM by at least 15% compared to the DNN model. Figure 3.3.1.c shows the activity map for the raw data and the two Denoising models.



	R1	R2	R3	R4	R5	R6
Raw	0.227	0.22	0.227	0.229	0.236	0.241
DNN	0.462	0.365	0.394	0.382	0.418	0.475
HDNN	0.564	0.444	0.47	0.471	0.481	0.565

Fig. 3.3.1.b Numerical results of simulation of the third dataset

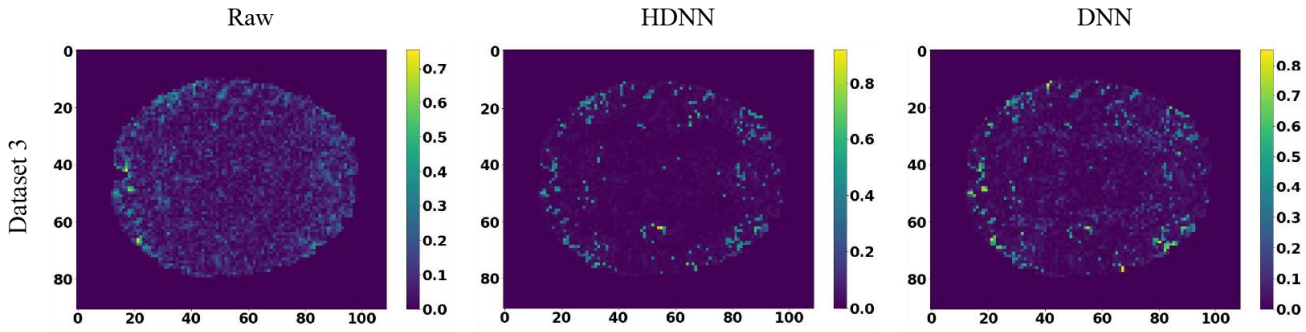


Fig. 3.3.1.c Activity mask for simulation of third dataset: derived from 1) Raw data 2) DNN model 3) HDNN model

In addition, the mean correlation changes of GM and non-GM data for HDNN model and DNN are examined, which can be seen in Figure 3.3.1.d. It confirms the proposed model could have a better performance than the DNN model in increasing the GM data correlation. On the other hand, also was able to reduce the correlation of non-GM data significantly compared to the DNN model.

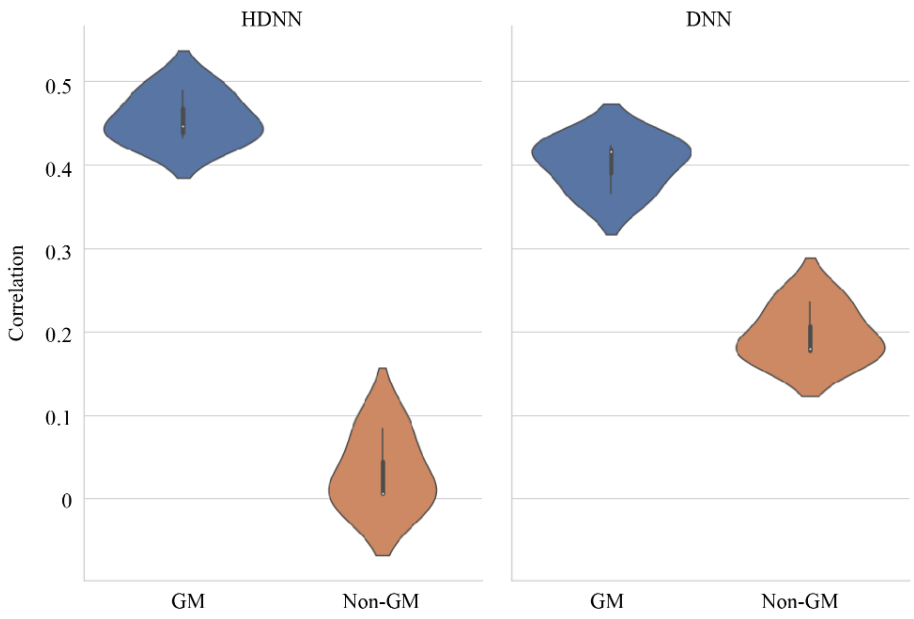


Fig. 3.3.1.d Mean correlation changes of GM and non-GM data using HDNN and DNN models

3.3.2 Real Data Analysis

In the previous section, the parameters related to the neural network model are tuned based on the simulated data. In this section, the performance of the proposed model for real data will be examined. According to Figure 3.3.2.a, it can be seen that the proposed model has presented an activity map with high contrast in GM areas. Tough the DNN model could increase the activity of GM regions, it was not able to reduce the non-GM correlation. In this section, as a general assessment, the average correlation of GM and non-GM data is shown in Figure 3.3.2.b. Like the simulated data analysis, using real data, HDNN is able to increase average GM correlation and decrease non-GM one significantly. As conclusion, our model fulfills the improvement in Denoising task-based fMRI compared to the prior model.

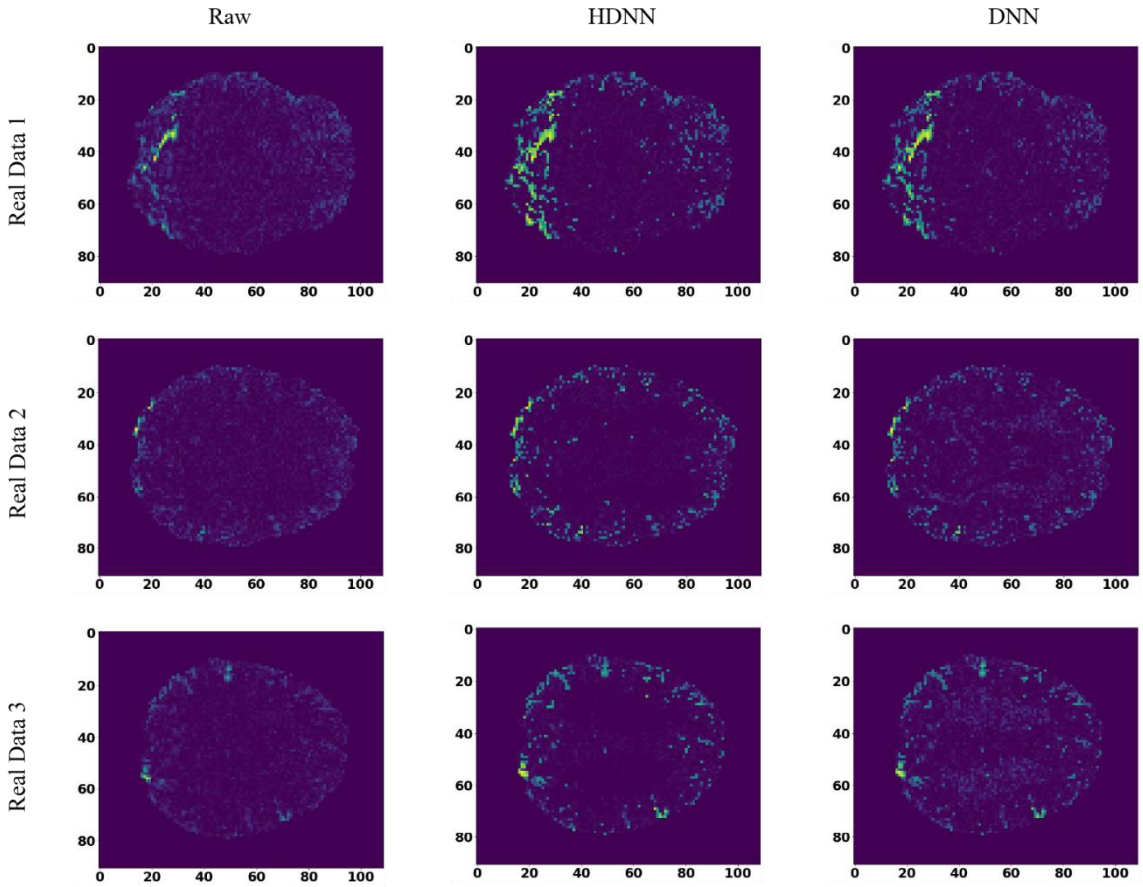


Fig. 3.3.2.a Activity map for a set of randomly selected real data

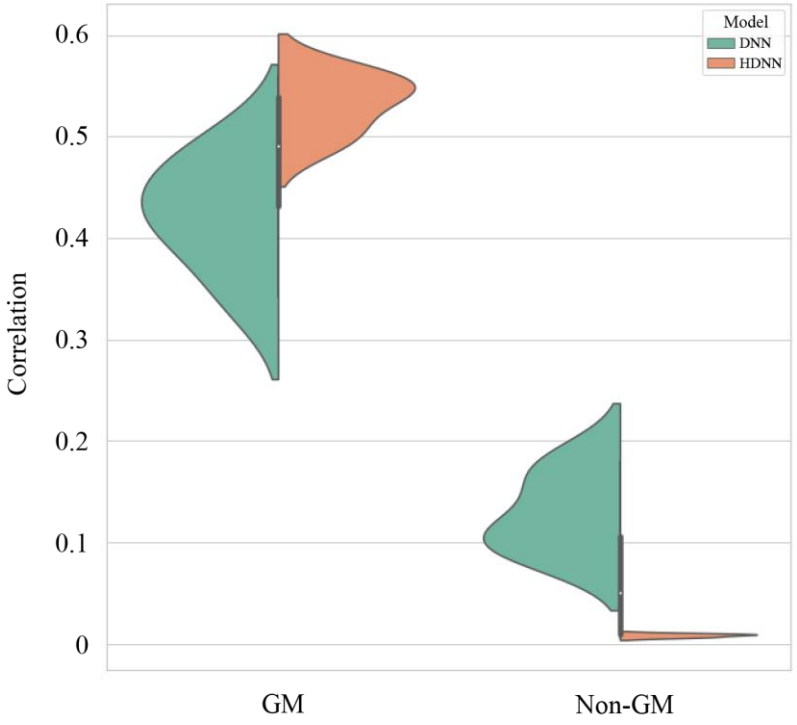


Fig. 3.3.2.b Mean correlation between GM and non-GM data using HDNN and DNN models

Chapter 4: Conclusion

Technology can assist doctors in making more accurate diagnoses and reducing the number of errors they make. A hybrid deep neural network (HDNN) is deployed in this article to minimize the noise in task-based fMRI data. The proposed model was created to increase the DNN model's performance [1]. Like the DNN paradigm, this network has a sequential structure. To reduce noise, the model employs three common layers and four non-common layers. The customized model which is discussed in this study would help noise reduction in task-based FMRI data more than the traditional approaches and even more than the DNN model which was our reference. Based on the results obtained in the proposed model, it is definitely determined that better outcomes against noise have been acquired. The focus of the model is on both GM and non-GM correlations, unlike the previous model, which despite its efforts on both GM and non-GM data, has had remarkable results mainly in the GM area. HDNN was able to achieve at least a 5% increment in average GM correlation which could be the goal of this study. It should be noted that the so-called proposed model has more complexity than the DNN model due to its hybrid architecture which would be the tradeoff between the complexity and performance. As a traditional approach, using GLM, the design matrix is obtained by convolving the binary task with the canonical HRF for both simulated and actual fMRI data Since the sole purpose of this research was to analyze the effect of modifications in network architecture on the output data, we followed the same approach. However, the further study to improve the Denoising algorithm can be the investigation of an untraditional way to calculate a more accurate design matrix, which is one of the main parameters of the network structure.

References

[1] Zhengshi Yang a, Xiaowei Zhuang a, Karthik Sreenivasan a, Virendra Mishra a, Tim Curran b, Dietmar Cordes, A robust deep neural network for denoising task-based fMRI data: An application to working memory and episodic memory, Medical Image Analysis 60 (2020) 101622

[2] Z.-P. Liang and P. Lauterbur, *Principles of magnetic resonance imaging: a signal processing perspective*. 2000.

[3] K. L. Miller, B. A. Hargreaves, J. Lee, D. Ress, R. C. DeCharms, and J. M. Pauly, "Functional brain imaging using a blood oxygenation sensitive steady state," *Magn. Reson. Med.*, vol. 50, no. 4, pp. 675–683, Oct. 2003.

[4] P. Bandettini, "Seven topics in functional magnetic resonance imaging," *J. Integr. Neurosci.*, 2009.

[5] K. K. Kwong et al., "Dynamic magnetic resonance imaging of human brain activity during primary sensory stimulation.," Proc. Natl. Acad. Sci. U. S. A., vol. 89, no. 12, pp. 5675–5679, 1992.

[6] S. Ogawa, T.-M. Lee, A. S. Nayak, and P. Glynn, "Oxygenation-sensitive contrast in magnetic resonance image of rodent brain at high magnetic fields," Magn. Reson.Med., vol. 14, no. 1, pp. 68–78, Apr. 1990.

[7] (https://www.fil.ion.ucl.ac.uk/spm/)

[8] R. Poldrack, J. Mumford, and T. Nichols, *Handbook of functional MRI data analysis*. 2011.

[9] A. Gholipour, N. Kehtarnavaz, R. Briggs, M. Devous, and K. Gopinath, "Brain functional localization: A survey of image registration techniques," *IEEE Trans. Med. Imaging*, vol. 26, no. 4, pp. 427–451, 2007.

[10] K. J. Friston, a. P. Holmes, K. J. Worsley, J.-P. Poline, C. D. Frith, and R. S. J. Frackowiak, "Statistical parametric maps in functional imaging: A general linear approach," *Hum. Brain Mapp.*, vol. 2, no. 4, pp. 189–210, 1995.

[11] https://github.com/neurodebian/spm12/blob/master/spm_hrf.m

[12] Martin A. Lindquist* , Ji Meng Loh, Lauren Y. Atlas, and Tor D. Wager, Modeling the Hemodynamic Response Function in fMRI: Efficiency, Bias and Mis-modeling, , Columbia University, New York, NY, 10027, Department of Psychology, Columbia University, New York, NY, 10027, Neuroimage. 2009 March; 45(1 Suppl): S187–S198. doi:10.1016/j.neuroimage.2008.10.065

[13] https://www.humanconnectome.org