Vector Auto-Regressive Deep Neural Network: A Data-Driven Deep Learning-Based Directed Functional Connectivity Estimation Toolbox

Takuto Okuno* and
Alexander Woodward for the Alzheimer's Disease Neuroimaging Initiative†

Connectome Analysis Unit, RIKEN Center for Brain Science, Wako, Japan

An important goal in neuroscience is to elucidate the causal relationships between the brain's different regions. This can help reveal the brain's functional circuitry and diagnose lesions. Currently there are a lack of approaches to functional connectome estimation that leverage the state-of-the-art in deep learning architectures and training methodologies. Therefore, we propose a new framework based on a vector auto-regressive deep neural network (VARDNN) architecture. Our approach consists of a set of nodes, each with a deep neural network structure. These nodes can be mapped to any spatial sub-division based on the data to be analyzed, such as anatomical brain regions from which representative neural signals can be obtained. VARDNN learns to reproduce experimental time series data using modern deep learning training techniques. Based on this, we developed two novel directed functional connectivity (dFC) measures, namely VARDNN-DI and VARDNN-GC. We evaluated our measures against a number of existing functional connectome estimation measures, such as partial correlation and multivariate Granger causality combined with large dimensionality counter-measure techniques. Our measures outperformed them across various types of ground truth data, especially as the number of nodes increased. We applied VARDNN to fMRI data to compare the dFC between 41 healthy control vs. 32 Alzheimer's disease subjects. Our VARDNN-DI measure detected lesioned regions consistent with previous studies and separated the two groups well in a subject-wise evaluation framework. Summarily, the VARDNN framework has powerful capabilities for whole brain dFC estimation. We have implemented VARDNN as an open-source toolbox that can be freely downloaded for researchers who wish to carry out functional connectome analysis on their own data.

Keywords: vector auto-regressive deep neural network, directed functional connectivity, fMRI, Granger causality, Alzheimer's disease

INTRODUCTION

There is a rich history of structural and functional connectome analysis of neuroimaging data of the whole brain, such as those acquired from MRI (magnetic resonance imaging). One important goal is to study the causal relationship between activity in different brain regions and the brain's functional circuitry. The estimation of the functional connectome is an important step for understanding brain function (Gao et al., 2011; Krueger et al., 2011; Hahn et al., 2019) and for the diagnosis of brain lesions (Dauwels et al., 2010). In the literature, there is a notable lack of data-driven approaches to functional connectome estimation that leverage the advantages of the current state-of-the-art deep neural network architectures and training methodologies. Such approaches should have an advantage in terms of representational power and trainability. Some recent examples of neural network-based measures of GC have been proposed, such as neural network GC (NN-GC) (Montalto et al., 2015), RNN-GC (Wang et al., 2018), Echo State Network GC (ES-GC) (Duggento et al., 2019), and DNN-GC (Chivukula et al., 2018). Accordingly, we propose a new approach for whole-brain analytics based on a *vector auto-regressive deep neural network* (VARDNN) architecture that can deal with a large number of time series. The model comprises a set of nodes with a deep neural network (DNN) structure. Each node maps with arbitrary divisions of space, such as an anatomical brain region from where a representative neural signal (time-series) can be estimated. At each time step, a node outputs a signal by considering exogenous signals into the system, signals from all other nodes, and feedback from the node back to itself. The model can be trained on time series data of brain activities and recordings of arbitrary exogenous inputs. Functional connectome measures can be derived from the model after the training process. A wide variety of data can be used for training, such as fMRI BOLD or calcium imaging data; our approach is fully data-driven. We seek a non-parametric model that not only fits the data but also describes important dynamical properties and can complement hypothesis-driven approaches (Valdes-Sosa et al., 2011). We defined two types of *directed functional connectivity* (dFC) measures based on the VARDNN, namely VARDNN-DI (Directional Influence, our new measure) and VARDNN-GC (based on Granger causality).

Many types of functional connectome analysis algorithms are currently used. *Functional connectivity* (FC), which is simply calculated by the correlation coefficient of *pairwise* time-series, is the most basic measure to analyze brain region relationships. This term was originally defined as the temporal correlation of a neurophysiological index measured in different brain areas (Gerstein et al., 1989; Friston et al., 1993). *Partial Correlation* (PC) is a *conditional* version of correlation. This is calculated by the correlation of regression residuals with other time-series. When two factors show a strong correlation, it may be necessary to consider the presence of a third factor. PC can remove this kind of third factor effect and show a normalized correlation. FC based on both correlation and partial correlation only consider the *zero-lag* relationship between signals. On the other hand, *Granger causality* (GC) is based on a vector auto-regression (VAR) scheme and is categorized as a *predictive* type measure. The original formulation of GC was for the *pairwise* case but has since been extended to the *conditional* and *multivariate* cases (Granger, 1969; Geweke, 1982; Goebel et al., 2003). Furthermore, an information-theoretic approach to understanding causal relationships, called *transfer entropy* (TE), has been developed (Schreiber, 2000). This has been shown to be equivalent to GC when the time series dynamics under study are normally distributed (Barnett et al., 2009). GC and TE are directional, thus these measures are categorized as *directed functional connectivity* (dFC) (Friston et al., 2013), while FC and PC are categorized as *undirected functional connectivity*.

Some measures depend on a linear regression scheme, thus their coefficients and residuals are calculated by ordinary least squares (OLS). But OLS has several drawbacks: if the predictor variables (P) are larger than the observed data (N), the solution faces overfitting ($P >> N$ problem). Furthermore, if the observed data are strongly correlated with each other, the Gram matrix cannot be inverted and OLS cannot calculate the correct solution (multicollinearity problem). In the case of analyzing functional MRI data, sometimes several regions have strong correlations, or the observed data may be smaller than the region number. Here, functional connectome analysis algorithms face several OLS problems. To avoid these situations, countermeasure techniques can be applied. For example, PC has been combined with a L1-norm regularized regression scheme (SPC-EN) (Ryali et al., 2012), and multivariate GC has been combined with Principal Component Analysis (PCA-cGCM) (Zhou et al., 2009) to reduce the data dimension and transform the data into an orthogonal vector space.

Accurate estimation of the brain's functional connectome is an important topic for neuroscience. One challenge is that the estimated functional connectome values are not consistent across analysis algorithms, which can cause confusion when interpreting the results of a particular measure. Dauwels et al. (2010) attempted to systematically investigate EEG (electroencephalogram) synchrony with a special focus on the early diagnosis of Alzheimer's disease (AD). They evaluated 29 types of functional measures, including the correlation coefficient and related measures, phase synchrony indices, GC, and stochastic event synchrony (SES) (Dauwels et al., 2008). Results showed that only two measures, GC and SES, were able to convincingly distinguish MCI (mild cognitive impairment) patients from age-matched controls. Another investigation was performed by Smith et al. (2011). They evaluated 20 functional methods, including the correlation coefficient and related measures, mutual information (Shannon, 1948), GC, the LiNGAM algorithm (Shimizu et al., 2006), and several Bayes Net modeling algorithms. Twenty-eight types of synthetic fMRI BOLD (blood oxygen level dependent) signals were generated by the DCM model (Friston et al., 2003), and all methods were evaluated for their estimation of the ground truth connectivity. In the results, the PC regularized inverse covariance estimation, and several Bayes Net methods showed the highest sensitivity. Prando et al. (2020) also evaluated the performance of functional connectome estimation using a number of methods. They compared sparse vs. spectral

DCM, multivariate GC, LiNGAM, and several Bayes Net modeling algorithms. In total, 9 methods were evaluated using synthetic fMRI BOLD signals generated by the DCM model. The results showed that the sparse DCM method gave the highest accuracy and multivariate GC was competitive. Overall, Granger causality-based measures appear to give consistent performance for functional connectome estimation from both empirical and synthetic signals. A purely data-driven approach makes less assumptions about the underlying system generating the dynamics; hence, it can be applied more broadly to different types of phenomena and is less computationally expensive.
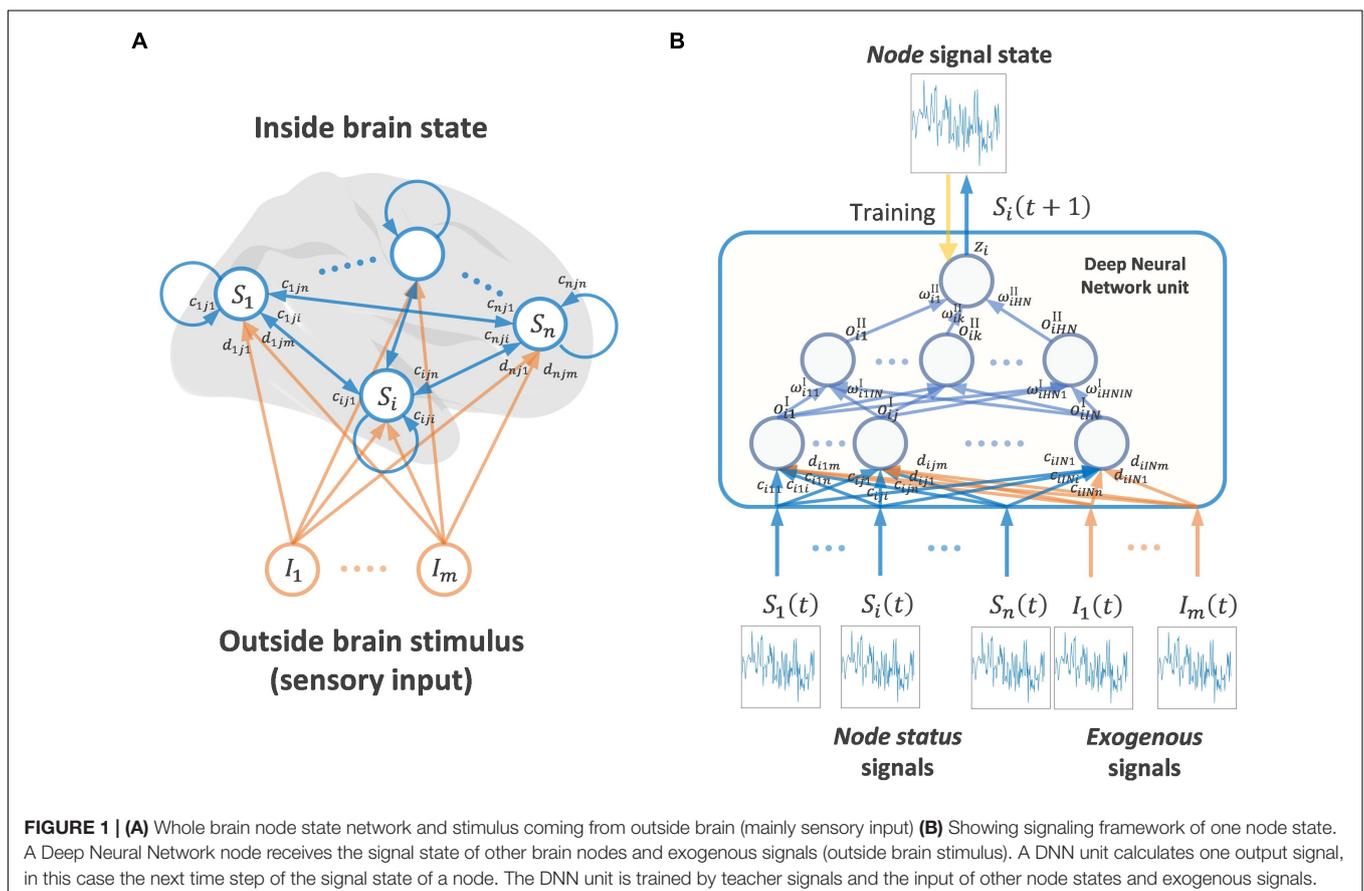
To evaluate our VARDNN approach we tested the performance of its dFC measures against 14 other analysis algorithms, including *zero-lag* and *predictive* types, and *pairwise* and *multivariate* strategies. Included in these algorithms, countermeasure techniques, such as PCA, Elastic Net (Zou and Hastie, 2005) and Partial Least Squares (PLS) (Wold et al., 2001) were combined with PC and multivariate GC. All measures were evaluated on biologically plausible synthetic signals generated by DCM and the reduced Wong-Wang neural mass model (Wong and Wang, 2006; Deco et al., 2013). In particular, we found that both measures, especially VARDNN-DI, were able to consistently provide competitive and top-ranking performance, especially when the size of the number of nodes increased. Furthermore, we evaluated VARDNN performance by analyzing empirical fMRI BOLD data. Healthy control and Alzheimer's disease fMRI BOLD data were acquired from the *Alzheimer's Disease Neuroimaging Initiative* (ADNI2) database, and functional connectomes were calculated by each of the analytical methods studied. Functional connectome values were then used to compare group separation ability of Alzheimer's disease within a subject-wise evaluation framework. Functional connectomes estimated by the VARDNN measure was able to capture the causal relationship between brain regions and showed a significant difference between healthy control and Alzheimer's disease data.

MATERIALS AND METHODS

Vector Auto-Regressive Deep Neural Network Architecture

Figure 1 shows a schematic diagram of the VARDNN framework. The architecture consists of a set of nodes that can be assigned to different regions of the brain from which experimental signals are recorded. The node state vector is $S(t) = (S_1(t), \cdots, S_n(t))$, where $S_i(t) \in R$, $i = 1,..,n$, represents the ith node's state at discrete time $t = 1, 2,\ldots$. All nodes are part of a fully connected network structure, receiving input from every other node. Additionally, each node receives exogenous signals from outside of the brain, $I_q(t) \in R$, $q = 1,\ldots,m$, expressed as an exogenous signal vector $I(t) = (I_1(t), \cdots, I_m(t))$. These signals can arbitrarily represent any external data derived

FIGURE 1 | (A) Whole brain node state network and stimulus coming from outside brain (mainly sensory input) **(B)** Showing signaling framework of one node state. A Deep Neural Network node receives the signal state of other brain nodes and exogenous signals (outside brain stimulus). A DNN unit calculates one output signal, in this case the next time step of the signal state of a node. The DNN unit is trained by teacher signals and the input of other node states and exogenous signals.

from an experimental paradigm that would be received as sensory information by the subject. All nodes contain a DNN unit with trainable weights across two hidden layers. For a node i, $c_{i,j,p}, d_{i,j,q} \in \mathbf{R}$ are the connection weights between its first hidden layer neurons $x_{i,j}$, $j = 1,\ldots,IN$, and the input from every other deep neural network unit node $p = 1,\ldots,n$ (including its self-connection), or from the exogenous signal inputs ($q = 1,\ldots,m$), respectively. At each time step, signals pass between nodes and are processed by the DNN units. Within a DNN, the output of the first hidden layer for node i is expressed as

$$O_i^{\mathrm{I}} = g(C_i S(t)^{\mathrm{T}} + D_i I(t)^{\mathrm{T}} + B_i^{\mathrm{T}})^{\mathrm{T}} \qquad (1)$$

where

$$C_i = \begin{pmatrix} c_{i,1,1} & \cdots & c_{i,1,n} \\ \vdots & \ddots & \vdots \\ c_{i,IN,1} & \cdots & c_{i,IN,n} \end{pmatrix}, \ D_i = \begin{pmatrix} d_{i,1,1} & \cdots & d_{i,1,m} \\ \vdots & \ddots & \vdots \\ d_{i,IN,m} & \cdots & d_{i,IN,m} \end{pmatrix},$$

$$B_i = (b_{i,1}, \cdots, b_{i,IN})$$

where $b_{i,j} \in \mathbf{R}$ are the bias values on the first hidden layer neurons and $g(x) = \max(0, x)$ is the non-linear ReLU activation function (Nair and Hinton, 2010; Glorot et al., 2011). O_i^{I} is passed to the second hidden layer neurons of node i, and its output is expressed as

$$O_i^{\mathrm{II}} = g(W_i^{\mathrm{I}} O_i^{\mathrm{I T}} + \beta_i^{\mathrm{I T}})^{\mathrm{T}} \qquad (2)$$

where

$$W_i^{\mathrm{I}} = \begin{pmatrix} \omega_{i,1,1}^{\mathrm{I}} & \cdots & \omega_{i,1,IN}^{\mathrm{I}} \\ \vdots & \ddots & \vdots \\ \omega_{i,HN,1}^{\mathrm{I}} & \cdots & \omega_{i,HN,IN}^{\mathrm{I}} \end{pmatrix}, \ \beta_i^{\mathrm{I}} = (\beta_{i,1}^{\mathrm{I}}, \cdots, \beta_{i,HN}^{\mathrm{I}})$$

$\omega_{i,j,k}^{\mathrm{II}} \in \mathbf{R}$ are the weight values of the second hidden layer neurons $k = 1,\ldots,HN$, and $\beta_{i,k}^{\mathrm{II}} \in \mathbf{R}$ are the biases. Finally, node i's DNN second hidden layer outputs are processed by an output layer neuron:

$$z_i(t) = W_i^{\mathrm{II}} O_i^{\mathrm{II} T} + \beta_i^{\mathrm{II}} \qquad (3)$$

where $W_i^{\mathrm{II}} = (\omega_{i,1}^{\mathrm{II}}, \cdots, \omega_{i,HN}^{\mathrm{II}})$ is the output neuron's weight vector, and $\beta_i^{\mathrm{II}} \in \mathbf{R}$ is its bias. Finally, the VARDNN's new state at $t+1$ for all nodes is taken as the value of each node's output:

$$S(t+1) = Z(t) = (z_1(t), \cdots, z_n(t)).$$

For our current implementation, we defined 2 hidden layers for each DNN unit in the VARDNN architecture. However, the DNN unit is not limited to 2 hidden layers, and other forms of DNNs can also be used in the framework. Our decision was based on this work's focus on analyzing whole brain fMRI BOLD type data, and 2 layers were empirically determined (see **Supplementary Figure 1** for a description of

the procedure). Although neural networks with single hidden layers are known as universal approximators (Cybenko, 1989; Hornik, 1989), a single hidden layer requires an exponential increase in the number of hidden neurons proportional to the problem size (Goodfellow et al., 2016; Liang and Srikant, 2016). Theoretically and experimentally, deep hidden layers with smaller numbers of neurons have shown higher accuracy than shallow (single) hidden layer networks (Delalleauand Bengio, 2011; Goodfellow et al., 2014).

VARDNN can learn to reproduce specific time series data through training on teacher data. The teacher data can be defined as $\mathbf{Teach}(t) = \left(S_1^{'}(t), \cdots, S_n^{'}(t), I_1^{'}(t), \cdots, I_m^{'}(t) \right)$, where $S_i^{'}$ and $I_i^{'}$ are the experimentally obtained signals of "neural activity" and (if any) exogenous inputs, respectively, for each node. Here, $t = 1,2,\ldots,L$, where L is the number of *frames* of data at a sampling period TR. The number of nodes, n, corresponds to the number of input time series. This number is arbitrary and should reflect some spatial sub-division of the brain, based on the nature of the teacher data. For example, if fMRI BOLD data is used, then one can average the voxel values within each anatomical region to generate a set of representative teacher signals.

Each node is independently trained with the aim to iteratively reduce the error between the VARDNN-generated $t+1$ state and the teacher data at time $t+1$, when the current state is set by the time t teacher data. With teacher data of length L frames, the training set for a node i comprises $\{\phi_i(1), \ldots, \phi_i(L-1)\}$, where $\phi_i(k) = \{\mathbf{Teach}(k), S_i^{'}(k+1)\}$ is an input/target pair, $k = 1,2,\ldots,L-1$. Even though these pairs are derived from time series data, they can be used independently. Thus, the shuffling and minibatch techniques can be applied in each training epoch. This feature of VARDNN is clearly different from recurrent neural network (RNN) architectures as VARDNN is simply trained by input/target patterns but not sequences.

Within each training epoch, for each training pair $\phi_i(k)$, the VARDNN node state and exogenous input are set to $\mathbf{Teach}(k)$, and the output scalar value $z_i(k)$ is calculated. Then, the error is evaluated as the difference $z_i(k) - S_i^{'}(k+1)$. The training error for node i is expressed over all $L-1$ training pairs as $\mathbf{Err}_i = y_i - \widetilde{z_i}$, where $y_i = (S_i^{'}(2), \ldots, S_i^{'}(L))$ and $\widetilde{z_i} = (z_i(1), \cdots, z_i(L-1))$. This error is used to drive the optimization procedure.

After a number of epochs, the training is completed for the current node, and this process is repeated for all nodes of the VARDNN system. Currently, the VARDNN framework was implemented in MATLAB using the Deep Learning Toolbox version 12.1. Training was optimized using Adam (Kingma and Ba, 2014) with L2 regularization for the hidden layers to avoid over-fitting.

Vector Auto-Regressive Deep Neural Network-Granger Causality Measure

Granger causality is a well-established method for the causal analysis of time series data, and the multivariate version (mvGC) has successfully been used to analyze fMRI BOLD signals

(Goebel et al., 2003; Menon and Uddin, 2010; Marinazzo et al., 2011; Wen et al., 2012). The traditional approach to multivariate Granger causality uses a vector auto-regressive model involving a state vector $S(t)$, time lag p, coefficient vector A_{ik}, and residual $\varepsilon_i(t)$. The state of node i can be described by the following model:

$$S_i(t) = \sum_{k=1}^{p} A_{ik} S(t-k)^{\mathrm{T}} + \varepsilon_i(t). \tag{4}$$

Removing the past dependency of node i on j, the model can be written as

$$S_{i\backslash j}(t) = \sum_{k=1}^{p} A_{ik\backslash j} S_{i\backslash j}(t-k)^{\mathrm{T}} + \varepsilon_{i\backslash j}(t). \tag{5}$$

Then the mvGC causal index is expressed as

$$\mathcal{F}_{j \to i} = \log\left(var\left(\boldsymbol{\varepsilon}_{i\backslash j}\right)/var\left(\boldsymbol{\varepsilon}_{i}\right)\right) \tag{6}$$

where $\boldsymbol{\varepsilon}_i$ and $\boldsymbol{\varepsilon}_{i\backslash j}$ are time series of residuals, and $var()$ denotes the statistical variance. For mvGC, the coefficients and residuals of the vector auto-regressive model are calculated by OLS. However, as mentioned in section "Introduction," neural network-based measures of GC have recently been proposed. VARDNN can also calculate this type of Granger causality. We simply use $\boldsymbol{Err}_i = \boldsymbol{y}_i - \widetilde{\boldsymbol{z}}_i$ of a single node i as the residuals $\boldsymbol{\varepsilon}_i$ and the error when removing the signal generated from node j from the optimization, $\boldsymbol{Err}_{i\backslash j} = \boldsymbol{y}_i - \widetilde{\boldsymbol{z}}_{i\backslash j}$ as the residuals $\boldsymbol{\varepsilon}_{i\backslash j}$. Then, the VARDNN-GC measure is defined as

$$\mathcal{F}_{j \to i}^{VARDNN-GC} = \log\left(var\left(\boldsymbol{Err}_{i\backslash j}\right)/var\left(\boldsymbol{Err}_{i}\right)\right) \tag{7}$$

In the context of neuroscience, the process of removing the signal from another node can be thought of as performing a simulated lesion (Honey et al., 2007; Alstott et al., 2009). This virtual impairment can allow us to understand the effect of possible lesions in terms of their (Granger causal) connectivity relationships. Therefore, VARDNN can also function as a lesion model.

Vector Auto-Regressive Deep Neural Network-Directional Influence Measure

Vector auto-regression is a well-known statistical approach to model the relationship between a scalar response and one or more explanatory variables. From this perspective, we can describe the state of a node i as

$$\begin{aligned} S_i(t) &= \alpha_{i,0} + \boldsymbol{\alpha}_i S(t-1)^{\mathrm{T}} + \varepsilon_i(t) \\ &= \alpha_{i,0} + \sum_{k=1}^{n} \alpha_{i,k} S_k(t-1) + \varepsilon_i(t) \end{aligned} \tag{8}$$

with state vector $S(t)$, linear relationship vector $\boldsymbol{\alpha}_i = (\alpha_{i,1}, \cdots, \alpha_{i,n})$ and a residual term $\varepsilon_i(t)$. Now,

if we set all inputs into the node as 1, in other words $S(t-1) = (1, \cdots, 1)$ we get

$$S_i^{\mathrm{one}} = \sum_{k=0}^{n} \alpha_{i,k} + \varepsilon_i(t) \tag{9}$$

If input to node j is set to 0, $S_{i\backslash j}^{\mathrm{one}}$ is expressed as

$$S_{i\backslash j}^{\mathrm{one}} = \sum_{k \neq j} \alpha_{i,k} + \varepsilon_i(t) \tag{10}$$

Then, the magnitude of the linear relationship, $\alpha_{i,j}$, or *directional influence* (DI), from node j to i is expressed as

$$\alpha_{i,j} = S_i^{\mathrm{one}} - S_{i\backslash j}^{\mathrm{one}} \tag{11}$$

The VARDNN formula is complex, but non-linearly extended from this VAR formulation. After training the VARDNN, the deep neural network weights of each node encode its non-linear relationships with all other nodes. In a similar manner to the linear case, we defined a new measure of *directional influence*, VARDNN-DI, that extracts the non-linear magnitude of causal relationships from the network weights. Here, the DI from node j to node i is defined as

$$\mathcal{F}_{j \to i}^{VARDNN-DI} = \left| z_i^{\mathrm{one}} - z_{i\backslash j}^{\mathrm{one}} \right|. \tag{12}$$

In this case, z_i^{one} is calculated by setting all inputs into the node to 1, that is $S(t) = (1, \cdots, 1)$ and $I(t) = (1, \cdots, 1)$, and passing these values through its DNN. $z_{i\backslash j}^{\mathrm{one}}$ is also calculated in the same way but with the connection weight from node j set zeros in the first hidden layer neurons of the DNN of node i. As a result, the learned non-linear relationship from node j to node i can be extracted.

Pre-processing of Vector Auto-Regressive Deep Neural Network Teacher Data

Although it is not a restriction, the VARDNN was designed to preferably deal with data in the range [0, 1]. In practice it may occur that the teacher signals are not normalized to this range. For example, resting state fMRI (rsfMRI) BOLD data has a normal distribution centered around zero (Chen et al., 2003). Thus, we need to transform experimentally derived time series data into the [0, 1] range. To do this, the mean and standard deviation of the data are first estimated:

$$\mu^* = \frac{1}{nL} \sum_{i=1}^{n} \sum_{t=1}^{L} S_i(t), \tag{13}$$

$$\sigma^* = \sqrt{\frac{1}{nL} \sum_{i=1}^{n} \sum_{t=1}^{L} \left(S_i(t) - \mu^*\right)^2}. \tag{14}$$

The values of a particular time series, mapping to a node i are normalized by

$$\theta_i(t) = \left(S_i(t) - \mu^*\right)/\sigma^*. \tag{15}$$

Then, a sigmoid function is used to transform the signal into the [0, 1] range:

$$S_i^{'}(t) = 1/\left(1 + e^{-\alpha\theta_i(t)}\right), \tag{16}$$

where the coefficient $\alpha = 1$ was used for our current experiments. This transformation was chosen as it deals with potential signal outliers (that, for example, can appear in fMRI BOLD data) better than a simple linear transformation (see **Supplementary Figure 2** for an evaluation of the effect of sigmoid transformation). Finally, the resulting set of time series, $S_i^{'} = \left(S_i^{'}(1), \cdots, S_i^{'}(L)\right)$, is used as teaching data for setting the node states of the VARDNN.

Performance Evaluation

We quantified the performance of the VARDNN-GC and VARDNN-DI measures over four different evaluations. The first evaluation confirmed the ability of VARDNN to learn and reproduce time series generated from uniformly distributed random numbers. The second evaluation estimated the ground truth directed graph from synthetic fMRI BOLD signals generated by the DCM, in order to clarify the relationship between network density and estimation accuracy. The third evaluation estimated the ground truth directed graph from synthetic neural activity signals generated by the reduced Wong-Wang model. This was done to clarify the relationship between node number and estimation accuracy. The DCM and reduced Wong-Wang models were chosen to generate biologically plausible synthetic neural signals for evaluation. The final evaluation was to measure VARDNN's performance on experimental data. Here, we evaluated Alzheimer's disease group separation ability using empirical rsfMRI BOLD signals. For each evaluation, we also compared our approach to other well-known functional connectome analysis algorithms, listed in **Table 1**. Of these algorithms, two are *zero-lag*-based algorithms: FC and PC. FC is the simplest measure of *pairwise* similarity between two time series, based on covariance. We used the corr() function from MATLAB (R2019b) for the comparison. PC is a conditional version of FC. It refers to the normalized correlation between two time series. We used the partialcorr() function of MATLAB to calculate the PC. Because PC relies on linear regression, it may face large dimensionality and multi-collinearity problems. Therefore, we implemented several countermeasure combined versions of PC, namely Principal Component PC (PC-PC), Elastic Net PC (EN-PC), and Partial Least Squares PC (PLS-PC). Elastic Net requires hyperparameters α and λ; we calculated these values from the minimum error of fivefold cross-validation using half of the sample data.

The *pairwise* GC and *multivariate* (or conditional) GC algorithms were implemented as MATLAB scripts. The GC time lag p parameter was empirically chosen as 3 for the evaluation.

Multivariate GC, implemented as a multivariate vector auto-regression (mVAR) model, attempts to detect causal effects by considering differences between the network with all nodes and the network with the removal of a single node. On the other hand, the classical pairwise model simply considers two-node interactions. The mVAR model is widely used in many different fields, but it is also expected to suffer from the problems of data with large dimensionality and multicollinearity—the same as for the PC case. Therefore, we also implemented several counter-measure combined mvGC versions, namely multivariate Principal Component PC (mPC-GC), multivariate Elastic Net GC (mEN-GC), and multivariate Partial Least Squares GC (mPLS-GC) for our comparison.

Partially Conditioned GC (PCGC) is Granger causality based on partial conditioning, proposed by Marinazzo et al. (2012). The MATLAB script of its implementation was obtained from the GitHub.[1] Parameter "ndmax" of the init_partial_conditioning_par_m() function was calculated based on node number, and the partial_CGC_fix_nd_m() function was used to acquire the dFC matrix.

The RNN-GC implementation was obtained from the GitHub.[2] The evaluation environment was built using Anaconda3,[3] Python-3.6, Keras-1.2.2, and Tensorflow-1.14.0.[4] The dFC matrix was estimated as the average of 8 runs of the RNN_GC() function. In the first evaluation, we chose 30 as the number of hidden neurons.

Transfer entropy (TE) is a powerful tool for detecting the transfer of information between nodes and is capable of discovering non-linear interactions. It has been proven that TE is equivalent to Granger causality for data that can be assumed to be drawn from a Gaussian distribution (Barnett et al., 2009). We used the MuTE implementation (Montalto et al., 2014) in MATLAB for our comparison. This implementation automatically chooses the time lag based on the Bayesian information criterion. MuTE has three entropy estimators: the linear estimator (LIN), binning estimator (BIN), and nearest neighbor estimator (NN). Either a uniform or non-uniform embedding scheme is used to derive the transfer of information. As per the author's results, BIN-NUE shows good estimation in both the linear and non-linear cases, but our initial tests found that estimating a dFC matrix from the signals generated by DCM was unsuccessful. Therefore, we chose the LIN-UE and NN-NUE algorithms for our comparison.

In sections "VARDNN Performance With Respect to Network Density" and "VARDNN Performance With Respect to Node Count," various measures were used to estimate the ground truth directed graph, and in order to quantify them, evaluation was performed by changing the acceptance threshold and drawing an ROC curve. The ground truth matrix and result matrix were compared across the maximum value of the result matrix to its minimum. From this changing threshold, true and false positive rates of the result matrix were detected, and a ROC curve was

---

[1]https://github.com/danielemarinazzo/PartiallyConditionedGrangerCausality
[2]https://github.com/shaozhefeng/RNN-GC
[3]https://www.anaconda.com/
[4]https://www.tensorflow.org/

TABLE 1 | Comparison algorithms.

Name	Abbreviation	Type	Strategy	Implementation
Functional connectivity (Correlation)	FC	Zero-lag	Pairwise	MATLAB function
Partial correlation	PC	Zero-lag	Multivariate	MATLAB function
Principal component + Partial correlation	PC-PC	Zero-lag	Multivariate	Our Script (MATLAB)
Elastic net + Partial correlation	EN-PC	Zero-lag	Multivariate	Our Script (MATLAB)
Partial least square + Partial correlation	PLS-PC	Zero-lag	Multivariate	Our Script (MATLAB)
Pairwise Granger causality	pwGC	Predictive	Pairwise	Our Script (MATLAB)
Multivariate Granger causality	mvGC	Predictive	Multivariate	Our Script (MATLAB)
Principal component + Multivariate Granger causality	mPC-GC	Predictive	Multivariate	Our Script (MATLAB)
Elastic Net + Multivariate Granger causality	mEN-GC	Predictive	Multivariate	Our Script (MATLAB)
Partial Least Square + Multivariate Granger causality	mPLS-GC	Predictive	Multivariate	Our Script (MATLAB)
Partial Conditioning Granger causality	PCGC	Predictive	Multivariate	Script (MATLAB)
Recurrent neural network Granger causality	RNN-GC	Predictive	Multivariate	Script (Python)
Linear estimator-uniform embedding (Transfer entropy)	LINUE-TE	Predictive	Multivariate	MuTE (MATLAB)
Nearest neighbor estimator-Non-uniform embedding (Transfer Entropy)	NNNUE-TE	Predictive	Multivariate	MuTE (MATLAB)

generated from (0,0) to (1,1). After completing the ROC curve an area under the curve (AUC) value could be obtained. An AUC value close to 1 means a better classifier, whereas an AUC value close to 0.5 means that it is a random classifier. In this way, the accuracy of each measure was quantified by the value of AUC and used for our evaluation.

Synthetic Data for Algorithm Evaluation
Uniformly Distributed Random Data
In section "VARDNN Trained on Random Independent Signals," 8-nodes ($i = 1...8$) of 100 length time series were generated by the Mersenne Twister (Matsumoto and Nishimura, 1998) pseudo random generator, with values in the range [0, 1] of the uniform distribution. The VARDNN was trained with these random signals to verify the DNN unit functionality of each node.

DCM Generated Data
In section "VARDNN Performance With Respect to Network Density," we evaluated the relationship between network density and ground truth network estimation from signals generated by an 8-node DCM model. Eight nodes were chosen based on the confirmation that the VARDNN operated as expected based on the results given in section "VARDNN Trained on Random Independent Signals." We designed 5 patterns of sparsely connected directed networks for generating synthetic fMRI BOLD signals with DCM. The network densities were 0.2, 0.25, 0.3, 0.41, and 0.5. Each DCM connectivity matrix A was derived from these network graphs. Each node had self-connections with a weight of 0.2, and inter-node connections had random weight values in the range [0.2, 0.5]. The DCM input connection matrix C was set to identity so that each node received endogenous fluctuations as random white noise to model the external environment. To generate synthetic fMRI BOLD signals, parameters of $T = 300$ (number of observations) and $TR = 2$ (repetition time) were used. Endogenous fluctuations for each node were generated by the spm_rand_mar() function, and fMRI BOLD signals were generated by the observer function spm_int_J() with spm_fx_fmri and spm_gx_fmri routines. In

addition to the sparse network densities, we also evaluated networks with dense connectivity at the levels found in anatomical studies of the mammalian brain, for example 97% of all possible cortico-cortical connections have been observed in mouse anatomical results (Gămănuţ et al., 2018). Here, all nodes were connected (giving a network density of 1.14 when including self-connections), with uniform random weight values of [−0.1, 0.1] for designating weak connections, and weight values of [0.2, 0.5] for strong connections. Due to the DCM formulation, we found that such fully connected networks easily exhibit divergence, and signals could reach a plateau value. By setting negative weights on some of the weak connections, a stable signal could be produced. We designed 5 patterns of directed networks with strong weight ($|a_{ij}| > 0.2$) network densities set to 0.05, 0.11, 0.16, 0.27, and 0.36. In total, 10 patterns of 8-node fMRI BOLD signals and exogenous signals were generated by DCM for the evaluation.

Reduced Wong-Wang Generated Data
In section "VARDNN Performance With Respect to Node Count," we evaluated the relationship between network size and ground truth network estimation from synthetic neural activity signals generated by the reduced Wong-Wang model implemented in The Virtual Brain (TVB) platform. We generated global coupling matrices, defined as C, for the reduced Wong-Wang model by down-sampling Allen's mouse structural connectome matrix (Oh et al., 2014; Melozzi et al., 2017) that was constructed from a number of anterograde tracer injections. We chose the reduced Wong-Wang model along with structural connectome data to produce biologically plausible dynamics for analysis. We wanted to evaluate causal measures in a more biologically plausible setting; real structural connectivity data could act as a useful proxy for defining a realistic ground truth network for our analysis. Furthermore, the reduced Wong-Wang model was chosen to explore network size over the DCM model, as the latter was found to be too computationally expensive when a large number of nodes were used. Node counts were 16, 32, 48, 64, 80, and 98. Similar to the DCM case, all nodes were

TABLE 2 | Demographics of HC and AD subjects.

Variables	AD (*n* = 32)	HC (*n* = 41)	*p*-value
Male/Female	14/18	17/24	0.851
Mean Age (SD)	72.4 (2.2)	72.1 (2.6)	0.454

connected with weight values in [−0.05, 0.01], with a fraction of 0.15 connections assigned a strong weight value ($|c_{ij}| > 0.2$). The TVB framework was run using Anaconda3 and Python-3.6. We made small modifications to TVB's reduced Wong-Wang implementation to obtain appropriate signals for the evaluation. The "state_variable_boundaries" constant was bound to the range [0, 1], which could clamp activity signals, and we modified this value [−2.5, 2.5] in order to preserve causal relationships. Because the reduced Wong-Wang model has an internal noise factor, $\eta_i(t)$, we wanted to use this noise signal as an exogenous input for the VARDNN. Therefore, both the endogenous noise and neural activity signals were recorded and exported for evaluation. In summary, 6 patterns were used to generate 16–98 node sets of neural activity and internal noise signals. These were generated at a sampling frequency of 64 Hz, a global coupling $G = 1$ set to provide sufficient dynamics between regions, and a noise amplitude $\sigma = 0.014$.

Empirical Data Used to Evaluate Vector Auto-Regressive Deep Neural Network Performance

Alzheimer's Disease Neuroimaging Initiative Database

Data used in the preparation of this article were obtained from the *Alzheimer's Disease Neuroimaging Initiative* (ADNI) database.[5] The ADNI was launched in 2003 as a public-private partnership, led by Principal Investigator Michael W. Weiner, MD. The primary goal of ADNI has been to test whether serial magnetic resonance imaging (MRI), positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and early Alzheimer's disease (AD). **Table 2** shows the demographics of healthy controls (HC) and AD subjects from the *ADNI2* subset of the database. The number of subjects with HC and AD were 41 and 32, respectively. Younger subjects with AD were preferred; HC of similar ages were selected.

Resting State fMRI Data Pre-processing

fMRI BOLD data from the *ADNI2* was scanned using a 3.0 Tesla Philips MRI scanner and saved in the DICOM format. The experimental settings of the scanner were: flip angle = 80, repetition time (TR) = 3,000 ms, echo time (TE) = 30 ms, pixel size = 3.3 × 3.3 mm, slice thickness = 3.3 mm, matrix size = 64 × 64 × 48, and frame length = 140. More information on the resting state parameters can be found on the ADNI website.

We converted the DICOM data to 4D NIfTI files using the dcm2nii tool (Li et al., 2016). Images were re-orientated so

---

[5] adni.loni.usc.edu

---

that the anatomical axial, sagittal, and coronal planes appeared correctly. Next, the CONN toolbox (Whitfield-Gabrieli and Nieto-Castanon, 2012) was used to pre-process the fMRI data. CONN registered NII images to the standard MNI brain space. Slice-timing correction was 1, 3, 5, . . ., 2, 4, and 6 for the Philips 3T MRI. We removed the first 10 frames of fMRI data, and the remaining data was smoothed by a FWHM of 5 mm. After pre-processing, the fMRI data was segmented using the CONN default atlas (132 ROIs), which is a combination of the FSL Harvard-Oxford atlas cortical and subcortical areas and the AAL atlas cerebellar areas. Voxels in each ROI were averaged to give representative fMRI BOLD signals for 132 regions over 130 frames.

Subject-Wise Evaluation Framework for Algorithm Comparison

We analyzed fMRI BOLD signals of HCs and ADs using 12 algorithms in **Table 1** except RNN-GC and NNNUE-TE to estimate connectivity between brain regions. This generated 132 × 132 sized functional connectome matrices for each of the 41 HC's and 32 AD's results. These two groups were compared using statistical testing. We performed the Lilliefors test to check the distribution of functional connectome matrices and the Wilcoxon rank sum test (Mann-Whitney *U*-test) to compare the two groups. From this comparison, we obtained a 132 × 132 *p*-value matrix describing the actual performance of lesion detection by each analysis algorithm. The top N_{rr} (= 30–300 step by 30) most different regional relationships between the HC and AD (that is, the set with the N_{rr} smallest *p*-values in the matrix) were then used to separate subjects to HC or AD group If an input functional connectome value (for a particular regional relationship) generated from new subject data was within the range [$-\beta\sigma$, $\beta\sigma$] (where β = 1.5–2.0 with a stepsize of 0.1) of the AD's functional connectome population distribution, it is counted as "unhealthy"; otherwise, it is counted as "healthy." This was done for each of the input functional connectome values in the N_{rr} most different regional relationships and a "healthy count" was acquired. The "health rating" for a subject is simply derived by dividing the health count by N_{rr}.

$$\text{Health rating} = \frac{\text{Healthy count } (\text{of } N_{rr} \text{ regional relationships})}{N_{rr}} \quad (17)$$

Two thresholds of N_{rr} and β were swept over for the subject-wise evaluation. From this the ROC curves and area under the curves (AUCs) of a total of 60 conditions were calculated to quantify and compare the performance of each of the studied analysis algorithms. Note that this subject-wise evaluation framework cannot be used as a reliable diagnostic technique if the subject number is not large enough—which we realize is the case for the dataset we used in this section. In such cases, cross-validation will show different *p*-value matrices and top N_{rr} regional relationships and give unstable diagnosis results. However, if there are enough subjects, we believe the presented statistical framework can be reliably applied for AD diagnosis.

Statistical Analysis Methods

To test the obtained AUC values, we used Kruskal-Wallis ANOVA in MATLAB for significance testing, and the Steel test in R-4.0.3 for multiple comparisons. For normality testing of the functional connectome distribution, we used the Lilliefors test in MATLAB. To perform non-parametric testing between the HC and AD groups, we used the Wilcoxon rank sum test (Mann-Whitney U-test) in MATLAB. Statistical significance was set at $p < 0.05$.

RESULTS

Vector Auto-Regressive Deep Neural Network Trained on Random Independent Signals

Figure 2 shows the result of training the VARDNN on fully random and independent signals for 8-nodes. The following hyperparameters were used: training epochs = 1,000, and hidden layer 1 neuron count "hidden1" = 32 and layer 2 count "hidden2" = 22. The number of neurons per layer were empirically calculated by the method described in

Supplementary Figure 1. Before training, the VARDNN nodes output signals clearly different from the teacher data, but after training with the conditions depicted in **Figure 2A**, the VARDNN nodes were able to output signals similar to the teaching signal (**Figure 2**). Before training, the mean absolute error (MAE) was 0.456 and after training it became 0.018. **Figures 2C,D** show some performance measures for VARDNN training. The MAE and training time were measured while changing the number of nodes and signal length, with the number of training epochs = 1,000. For any node number the VARDNN converged well and MAE showed values less than 0.02. Training time took less than 1 min with 30 nodes and signals of length 100 frames, and around 9 min with 180 nodes and signals of length 1,000 frames on a computer with the following specifications: OS, Windows 10 Pro; CPU, AMD Ryzen Threadripper 3970X 32-Core Processor (3.70 GHz); Memory, 128 GB. This makes VARDNN training very practical for data needing 100 nodes or a similar order of magnitude (such is the case when considering the number of cortical regions defined in brain atlases). Next, we performed a simple check to confirm VARDNN's ability to estimate dFC; node 6′s signal was copied to node 2 and 4 at the next time step. The VARDNN was trained with new network (**Figure 2E**). Zero-lag analysis (FC) detected a strong relationship between nodes 2 and 4 (**Figure 2F**).
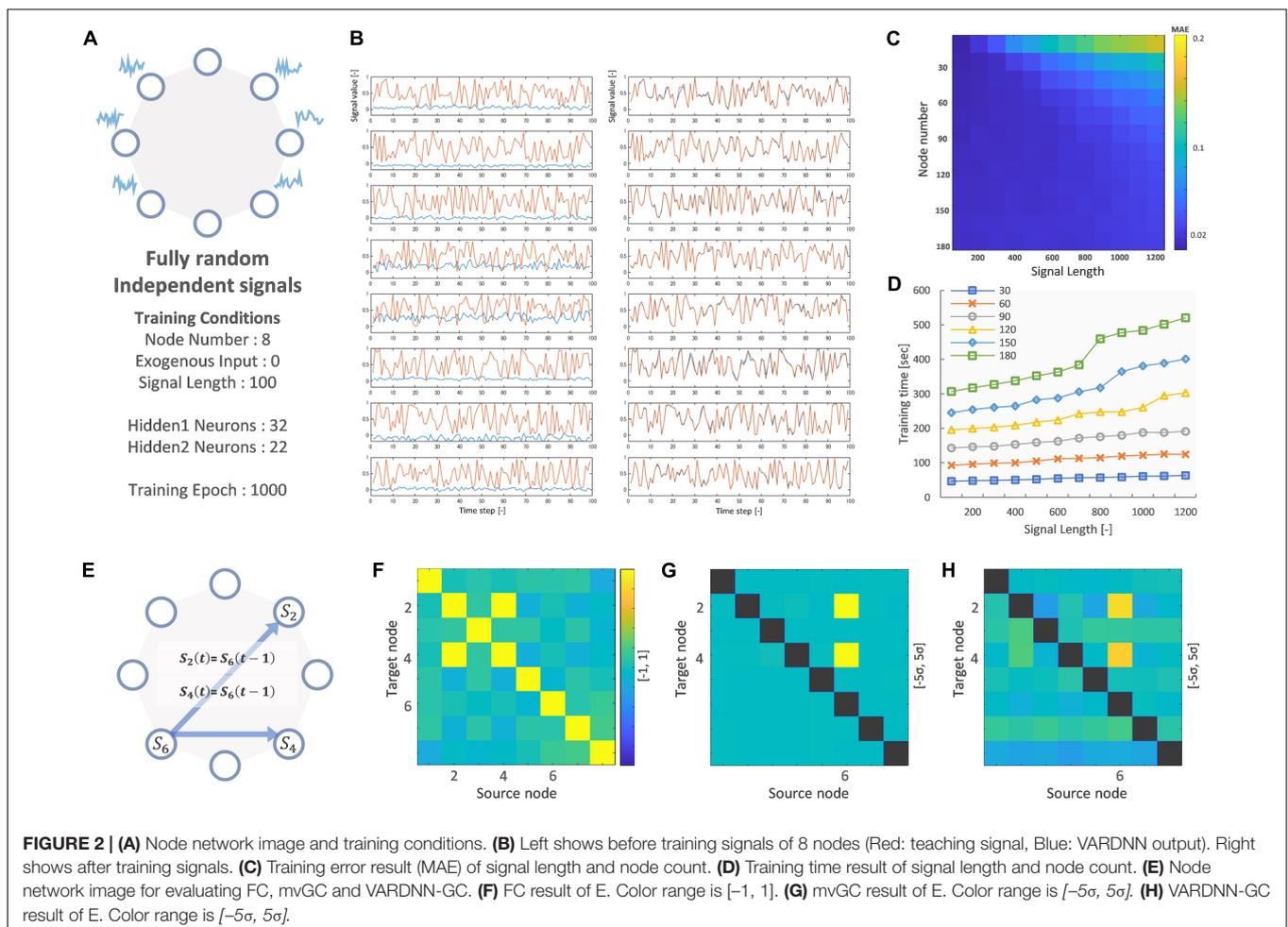

FIGURE 2 | (A) Node network image and training conditions. **(B)** Left shows before training signals of 8 nodes (Red: teaching signal, Blue: VARDNN output). Right shows after training signals. **(C)** Training error result (MAE) of signal length and node count. **(D)** Training time result of signal length and node count. **(E)** Node network image for evaluating FC, mvGC and VARDNN-GC. **(F)** FC result of E. Color range is [–1, 1]. **(G)** mvGC result of E. Color range is *[–5σ, 5σ]*. **(H)** VARDNN-GC result of E. Color range is *[–5σ, 5σ]*.

On the other hand, mvGC and VARDNN-GC detected strong causality from node 6 to nodes 2 and 4 (**Figures 2G,H**). Since Granger causality returns values that differ by two or more digits, the matrix is displayed as a deviation.

Vector Auto-Regressive Deep Neural Network Performance With Respect to Network Density

Figures 3, **4** A show evaluation results for the relationship between sparse pattern network density and ground truth estimation from signals generated by the 8-node DCM. The signal length was 300, and the trial number was $N = 8$. The VARDNN was trained with the synthetic BOLD signals and exogenous signals with training epochs = 1,000, hidden1 neurons = 55, and hidden2 = 37 (determined by the method described in **Supplementary Figure 1**). Both VARDNN-DI and VARDNN-GC show competitive results against conventional algorithms. In particular, VARDNN-DI (AUC: 0.92–0.94 across densities) shows a significantly higher AUC score compared to other algorithms, or a similar AUC score to mEN-GC (0.99–0.89), LINUE-TE (0.93–0.87), and higher as the density of strong connections increased. The zero-lag analysis algorithms, FC and PC families, showed similar results in all cases. mPLS-GC, mPC-GC, PCGC, RNN-GC, and NNNUE-TE sometimes showed a high AUC score but appeared unstable. mEN-GC, LINUE-TE, and mvGC showed higher AUC scores relative to the others, except for the VARDNN-based algorithms (especially in the high network density case). **Figures 4B**, **5**, show evaluation results for the relationship between strong weight ($| a_{ij} | > 0.2$) fully connected network density and ground truth estimation. Once again, VARDNN-based algorithms also showed competitive results against existing algorithms. As in the sparse network evaluation, VARDNN-DI (AUC: 1.0–0.94) showed a significantly higher AUC score over most algorithms and a similar AUC score to mEN-GC (1.0–0.92), LINUE-TE (1.0–0.92), and became higher as the density of strong connections increased. Increased network density causes synergistic signal activation and multicollinearity problems and VARDNN demonstrated robust capabilities for dealing with these issues.
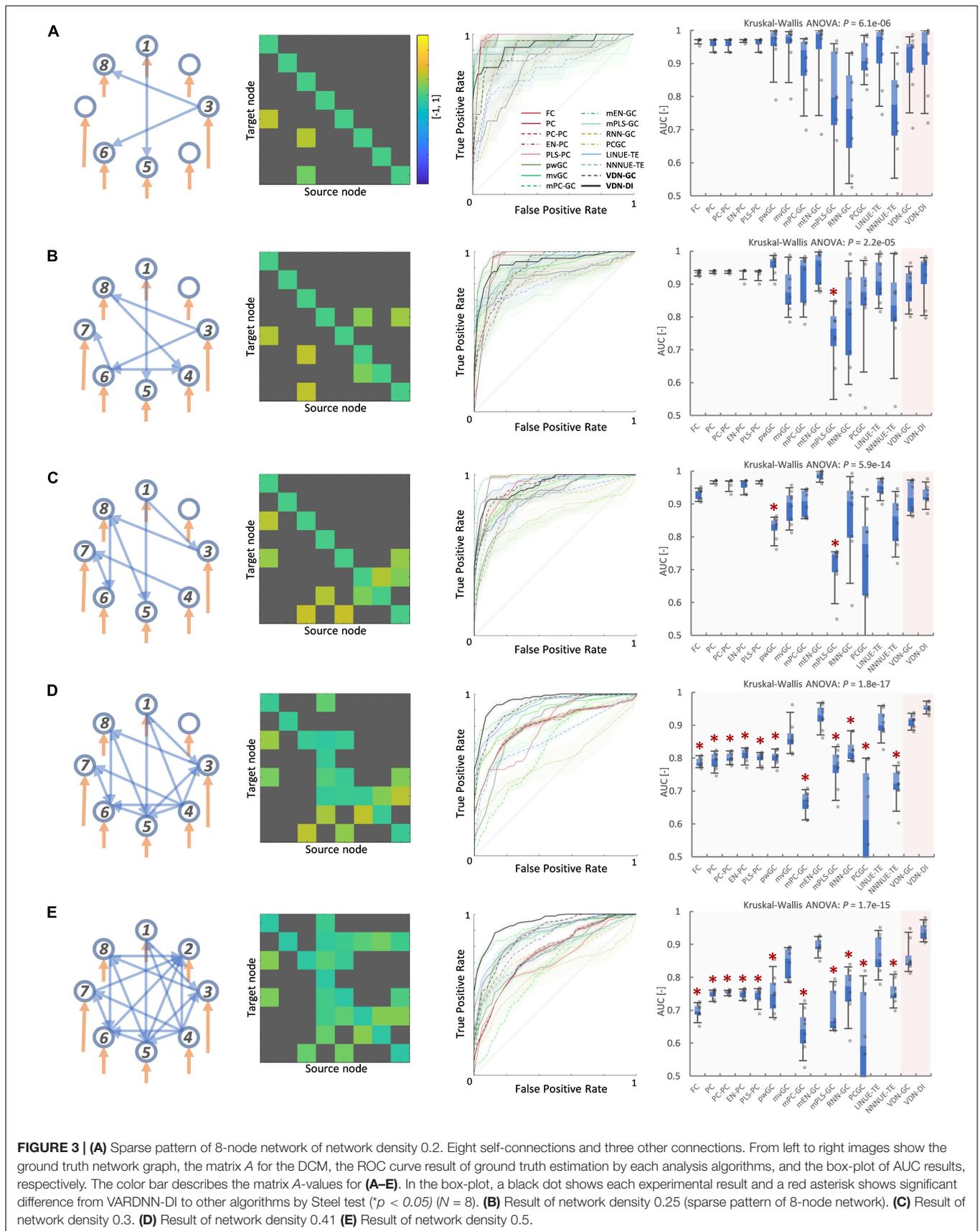
Computation time depends not only on the efficiency of the algorithm, but also on its implementation and runtime environment. However, as a basic guide we provide the computation times of the implementations we tested. The machine specification was the same as that described in section "VARDNN Trained on Random Independent Signals." Many algorithms, such as FC, PC, mvGC, pwGC showed an order of about 10^{-2} s for 8 nodes. LINUE-TE (0.05 s) showed an order of about 10^{-1} s. VARDNN-based algorithms (24 s) and NNNUE-TE (8 s) showed an order of about 10 s. On the other hand, RNN-GC (599 s) showed an order of more than 10^2 s. Because of this, RNN-GC was excluded from the evaluation in section "VARDNN Performance With Respect to Node Count" due to an excessively long calculation time in the large node case. In addition, NNNUE-TE was also excluded from the same evaluation due to a memory usage error.

Vector Auto-Regressive Deep Neural Network Performance With Respect to Node Count

Figures 6, **7** show evaluation results for the relationship between node count and ground truth estimation of node signals generated by the reduced Wong-Wang model. Conventional algorithms such as multivariate GC (AUC: 0.79–0.5), pairwise GC (AUC: 0.82–0.53), and LINUE-TE (AUC: 0.83–0.54) struggled when the number of nodes was large. As they are based on OLS, their sensitivity is reduced by the number of free parameters. Countermeasure combined mvGC showed a small advantage over standard mvGC. On the other hand, VARDNN-DI (AUC: 0.88–0.64) and VARDNN-GC (AUC: 0.74–0.66) showed better scaling performance. At node numbers above 48, VARDNN-DI and VARDNN-GC showed significantly higher AUC scores over LINUE-TE and the mvGC family, and above 64 was significantly higher than pwGC. The pwGC algorithm showed better scores than mvGC; this may be due to the number of free parameters. VARDNN-GC did not perform well for the smallest network size, but unlike the rankings in section "VARDNN Performance With Respect to Network Density," it outperformed VARDNN-DI for more than 64 nodes. Many zero-lag analysis methods did not perform well in any node count. This was because the sampling frequency (64 Hz) was too fast and correlations between nodes were lower than the fMRI BOLD signal repeat time (0.5 Hz). Overall, VARDNN showed a better AUC over linear regression results. This means VARDNN has the ability to be robust to redundancy or dealing with large dimensionality data. Interestingly, VARDNN shows a higher AUC over countermeasure techniques. This difference may be due to VARDNN's non-linearity.

Vector Auto-Regressive Deep Neural Network-Directional Influence and -Granger Causality Can Successfully Analyze Alzheimer's Disease

The rsfMRI data acquired from *ADNI2* was preprocessed then separated by the CONN default atlas, generating 132 regions and 130 frames of rsfMRI BOLD signals. These were analyzed by several analysis algorithms to obtain functional connectome matrices. VARDNN was trained with the empirical BOLD signals and randomly generated exogenous signals (uniform distribution in range [0, 1], acting as a noise factor to the algorithm), with training epochs = 1,000, hidden1 neurons = 34, and hidden2 = 23 (calculated by the method described in **Supplementary Figure 1**). **Figures 8A–D** show the mean functional connectome matrices for the AD and HC subjects. Most algorithms showed similar patterns for AD and HC, but PC, mvGC, and LINUE-TE did not (**Figure 8E**). The cosine similarities between AD and HC matrices were FC (0.965), PLS-PC (0.645), pwGC (0.976), VARDNN-DI (0.976), PC (0.11), mvGC (−0.025), and LINUE-TE (0.034). This meant that FC, pwGC, VARDNN, countermeasure combined PC and mvGC consistently captured meaningful regional relationships across the AD and HC subjects,
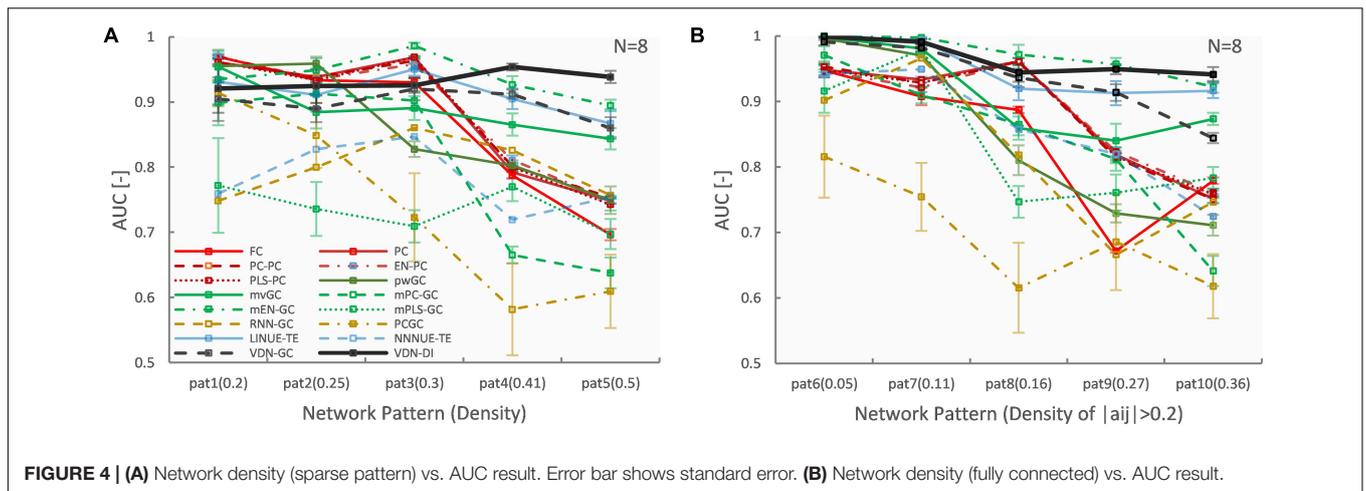
FIGURE 3 | (A) Sparse pattern of 8-node network of network density 0.2. Eight self-connections and three other connections. From left to right images show the ground truth network graph, the matrix A for the DCM, the ROC curve result of ground truth estimation by each analysis algorithms, and the box-plot of AUC results, respectively. The color bar describes the matrix A-values for (A–E). In the box-plot, a black dot shows each experimental result and a red asterisk shows significant difference from VARDNN-DI to other algorithms by Steel test (*$p < 0.05$) ($N = 8$). (B) Result of network density 0.25 (sparse pattern of 8-node network). (C) Result of network density 0.3. (D) Result of network density 0.41 (E) Result of network density 0.5.

FIGURE 4 | (A) Network density (sparse pattern) vs. AUC result. Error bar shows standard error. **(B)** Network density (fully connected) vs. AUC result.

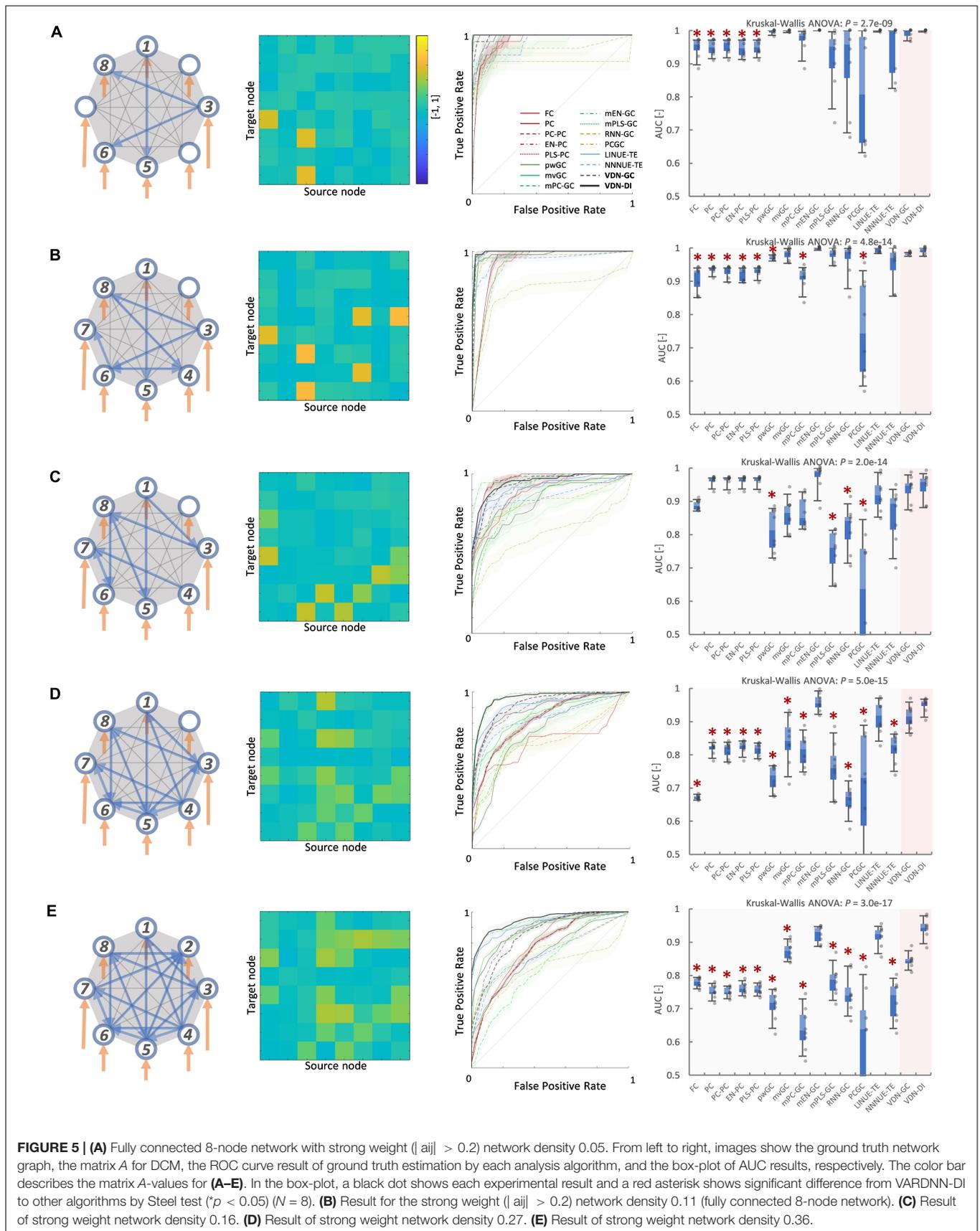but PC, mvGC, and LINUE-TE were clearly insensitive because of the large dimensionality problem.

To compare the values of functional connectome matrices (132×132) between the AD and HC groups we needed to test the normality of their distributions. FC values in the matrices for the HC group (21.3%) were significantly *not* Gaussian distributed ($p < 0.05$) (**Figure 8F**, left). VARDNN-DI values in the matrices for the HC group (74.1%) were significantly *not* Gaussian distributed ($p < 0.05$) (**Figure 8H** right). From these results, a non-parametric test was chosen as appropriate for comparing the AD and HC groups. We used the Wilcoxon rank sum test (Mann-Whitney U-test) for this comparison.

Figure 9 shows comparison results between the AD and HC groups for each analysis algorithm. VARDNN-DI detected that 12% of regional relationships were significantly affected ($p < 0.05$) by Alzheimer's disease; FC (10.5%), PLS-PC (5.5%), and pwGC (7.3%) were also able to detect a difference well. Even though these algorithms had functional connectome matrices showing high cosine similarity between AD and HC (**Figure 8E**), they could capture significant differences between the AD and HC groups. Next, we confirmed the consistency of our results with previous studies. **Table 3** shows the top 20 most different regions (in terms of dFC) between the AD and HC groups, calculated from VARDNN-DI.[6] Significantly different dFC were counted by brain region, and the total number of both sources and targets is displayed in the VDNdiTotal column in descending order. **Table 3** columns are sorted by the bold values in the VDNdiTotal column. In this result, VARDNN-DI was able to detect lesion effects of the Default Mode Network (DMN), including precuneus cortex, angular gyrus, and cingulate gyrus (posterior division). Additionally, supramarginal gyrus (posterior division) and intracalcarine cortex were also detected and matched well with previous studies. This result confirms that whole-brain screening with VARDNN-DI may be useful for further brain analysis. Both FC and pwGC detected some common effects within the DMN. Although these causal and

zero-lag analyses use different approaches, they were capable of showing similar results.

Figure 10 shows the result of Alzheimer's disease separation ability by our subject-wise evaluation framework. We tried to use two subject groups as a ground truth to quantify the separation ability of each algorithm. **Figure 10A** shows the top N_{rr} (= 100) most different regional relationships between the AD and HC groups by VARDNN-DI. From left to right, regional relationships were sorted in ascending order of p-value. As shown in this graph, the dFC distribution of AD was significantly different from that of HC. This means that connectivity is reduced in these regions. Then, we were able to use these N_{rr} regional relationships to separate AD and HC subjects. The functional connectome matrix was calculated from the rsfMRI BOLD signal obtained from a new subject, and the top N_{rr} regional relationships from the matrix were used to obtain their *health rating* (**Figure 10B**), which could be used to distinguish between AD ($N = 32$) and HC subjects ($N = 41$) (**Figure 10C**). ROC curves were generated from the subject distribution of AD and HC based on the *health rating* histogram. A "healthy" threshold to classify healthy or unhealthy was moved from 100 to 0% over the *health rating* range. In total, 60 conditions of the framework settings (N_{rr}:30–300 with step-size 30, β threshold: 1.5–2.0 with step-size 0.1) were tested for each algorithm. **Figure 10D** shows ROC curve results for each analysis algorithm, and **Figure 10E** shows the box-whisker plot of AUCs. VARDNN-DI (0.97) showed the highest AUC and PLS-PC (0.969) showed a competitive AUC result. These algorithms separated the AD and HC groups well. FC (0.897) and pwGC (0.888) (both pairwise strategies) showed similar cosine similarities and subject-wise evaluation results in TR = 3 s rsfMRI BOLD data. They in turn showed a similar ability for group separation. The mvGC family of algorithms and LINUE-TE showed unstable subject-wise evaluation results—they did not separate the two groups well. The PC family showed high AUC scores, but they were unstable over several thresholds. Also, their cosine similarity showed unstable results, i.e., their similarities were smaller than other algorithms (**Figure 8E**) and significantly different regional relations were fewer (**Figure 9**). For many brain
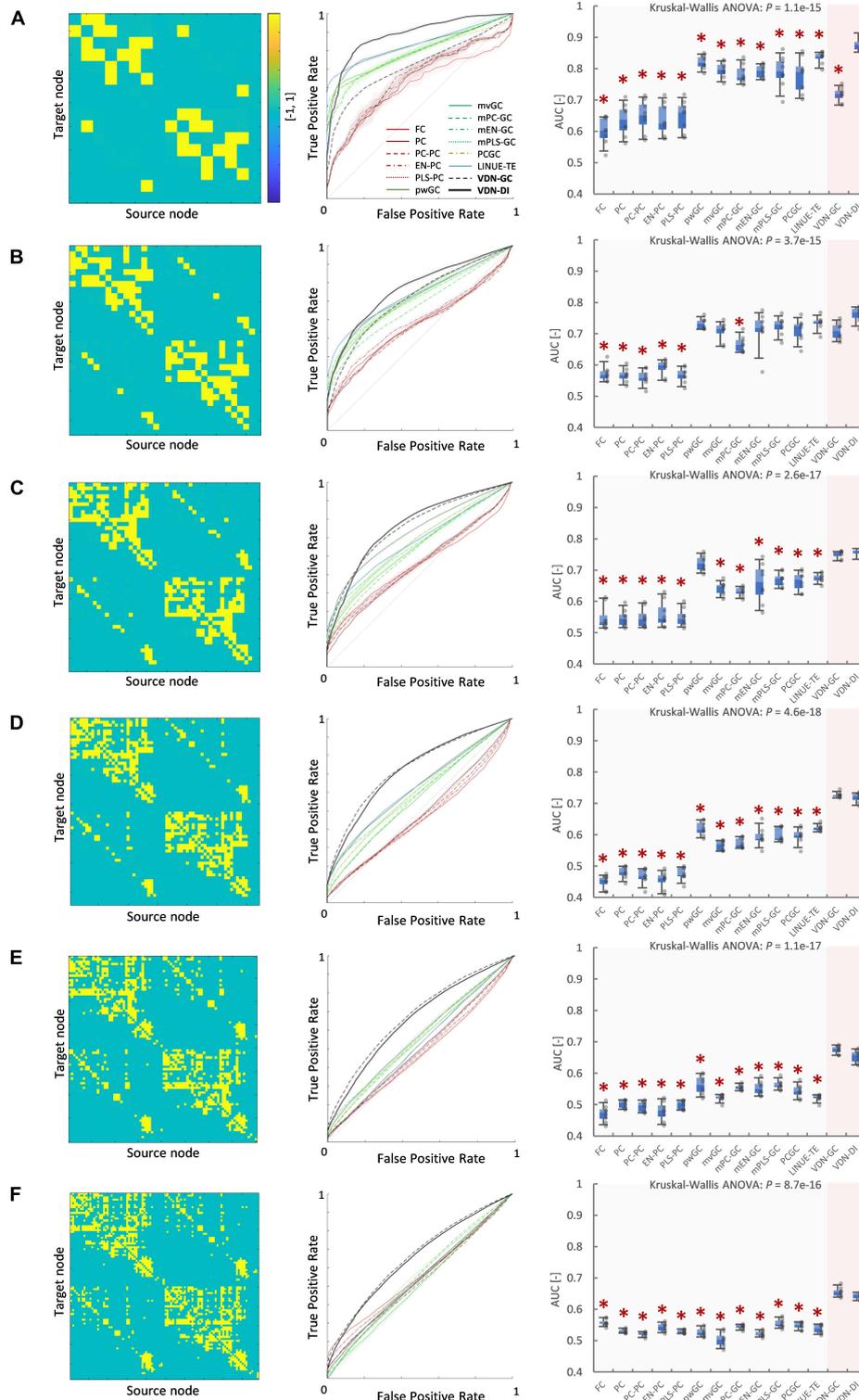
---

[6]VARDNN-DI was chosen over VARDNN-GC as we found it gave better diagnosis results, implying that VARDNN-GC's estimated significantly different regions were less accurate.

FIGURE 5 | (A) Fully connected 8-node network with strong weight (| aij| > 0.2) network density 0.05. From left to right, images show the ground truth network graph, the matrix *A* for DCM, the ROC curve result of ground truth estimation by each analysis algorithm, and the box-plot of AUC results, respectively. The color bar describes the matrix *A*-values for **(A–E)**. In the box-plot, a black dot shows each experimental result and a red asterisk shows significant difference from VARDNN-DI to other algorithms by Steel test (*p < 0.05) (N = 8). **(B)** Result for the strong weight (| aij| > 0.2) network density 0.11 (fully connected 8-node network). **(C)** Result of strong weight network density 0.16. **(D)** Result of strong weight network density 0.27. **(E)** Result of strong weight network density 0.36.
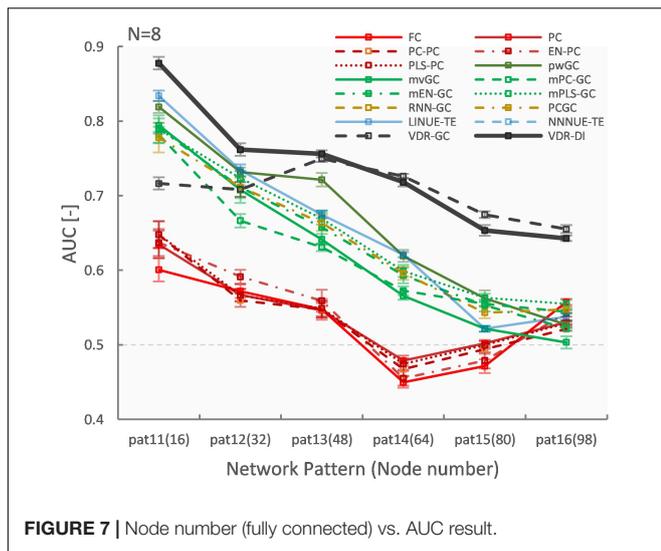
FIGURE 6 | (A) Fully connected 16-node network with strong weight (|c_{ij}| > 1) network density 0.15. Node signals were generated by the reduced Wong-Wang model from the TVB software. From left to right images show the global coupling matrix C for the reduced Wong-Wang model, the ROC curve result of ground truth estimation by each analysis algorithm, and the box-plot of AUC results, respectively. Color bar shows matrix C-value for **(A–F)**. In the box-plot, black dot shows each experimental result and red asterisks denote a significant difference from VARDNN-DI to other algorithms by the Steel test (*p < 0.05) (N = 8). **(B–F)** Results for the 32-node, 48-node, 64-node, 80 node, and 98-node networks, respectively.

FIGURE 7 | Node number (fully connected) vs. AUC result.

regions, the conditioning strategy shows very small differences around the zero value and the two group distributions were mixed. Even though some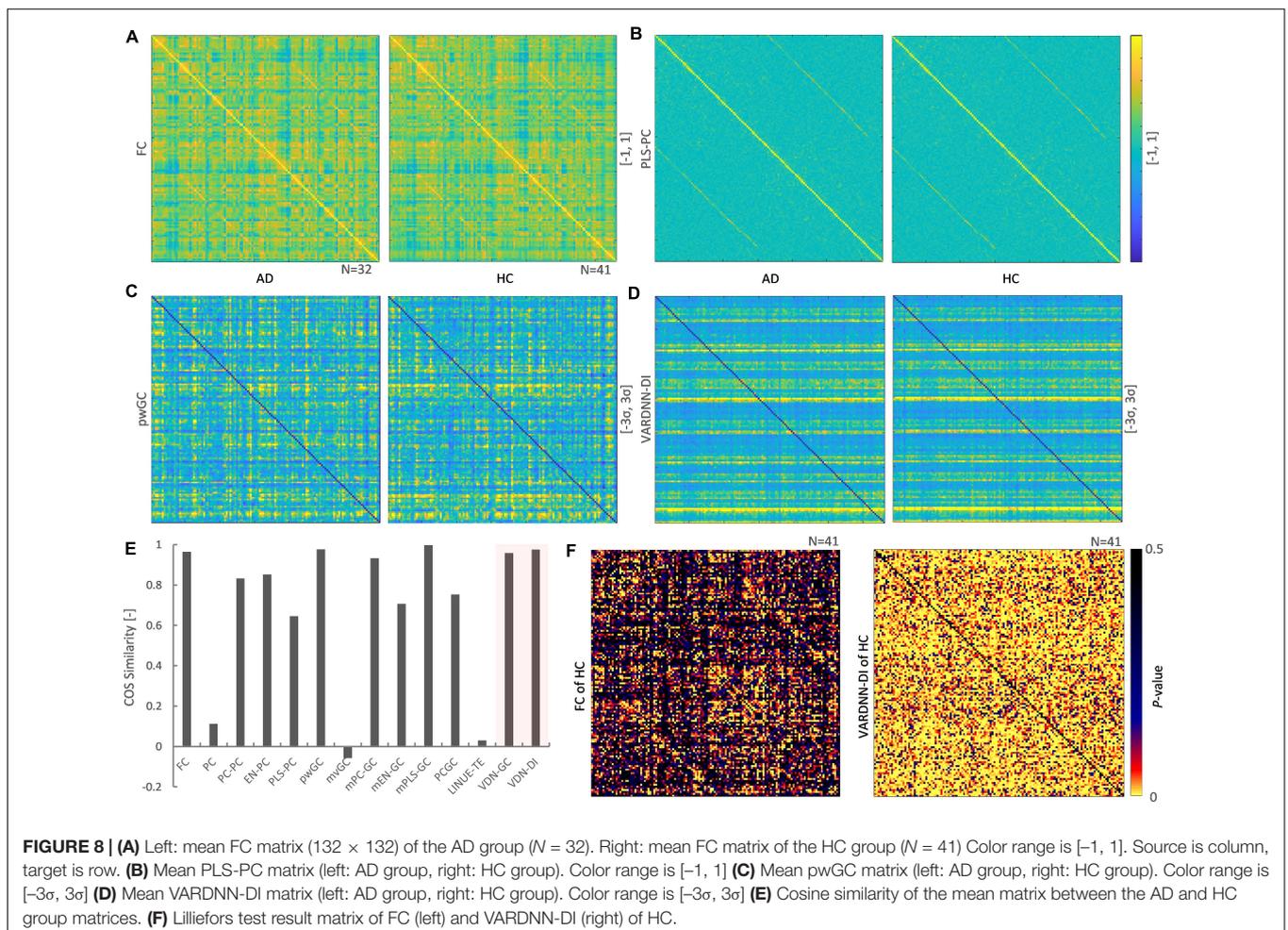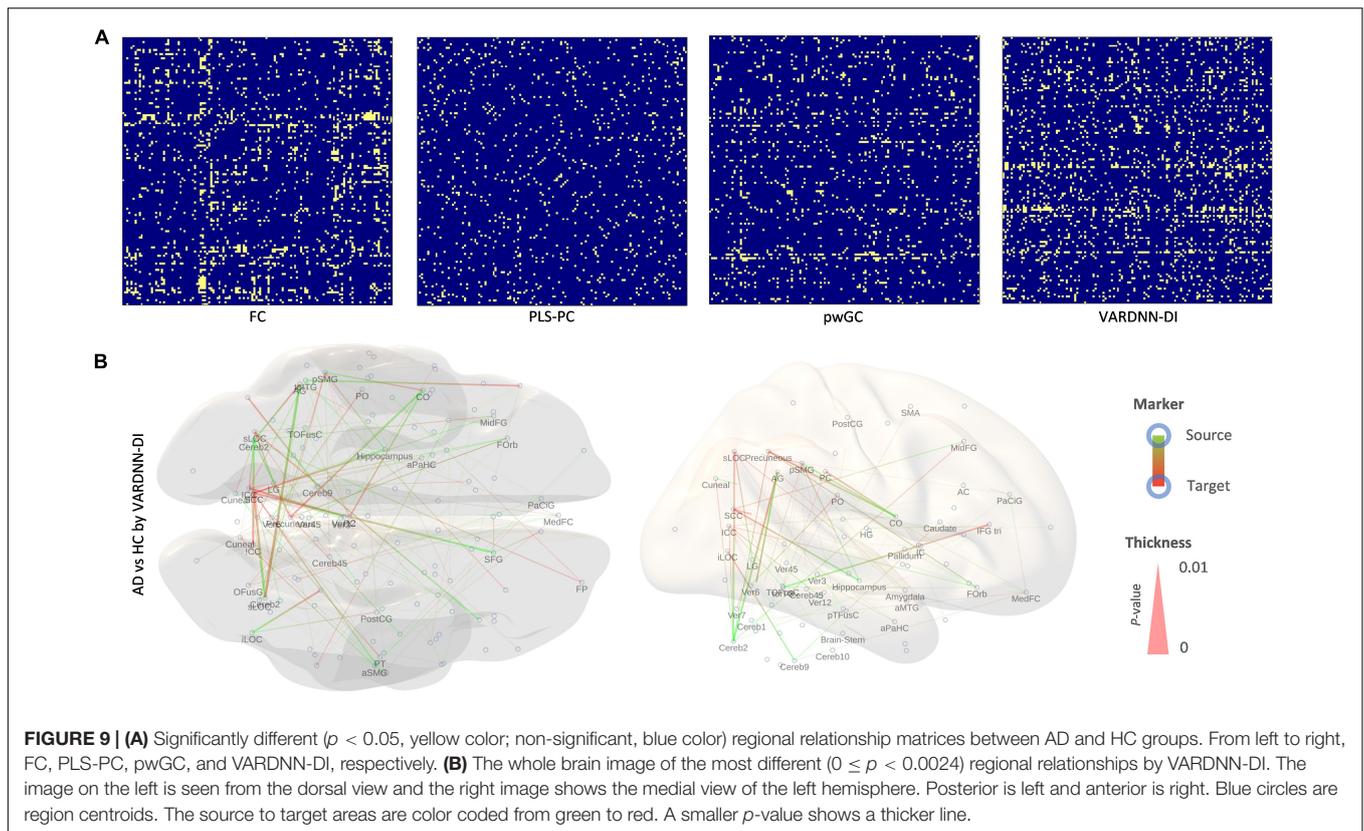 PC approaches were able to show good group separability, overall results were unstable. In summary, this demonstrates that VARDNN performs well with real-world, empirical rsfMRI BOLD data.

DISCUSSION

In this article, we presented VARDNN, a novel method for functional connectome estimation based on a vector auto-regressive deep neural network architecture. We proposed two VARDNN dFC measures that show good sensitivity to causal structures, namely VARDNN-GC and VARDNN-DI. In our experiments, the VARDNN was successfully trained with random uniformly distributed signals, synthetic fMRI BOLD signals that were generated by the DCM model, synthetic neural activity signals that were generated by the reduced Wong-Wang model, and empirical fMRI BOLD signals acquired from the *ADNI2* dataset. Functional connectome estimation of ground-truth synthetic fMRI BOLD signals from DCM was successful, and VARDNN-DI and VARDNN-GC showed competitive results with existing functional connectome estimation measures. In particular, VARDNN-DI showed a higher AUC score over other algorithms for the high network density case (**Figure 5**).

FIGURE 8 | (A) Left: mean FC matrix (132 × 132) of the AD group (N = 32). Right: mean FC matrix of the HC group (N = 41) Color range is [–1, 1]. Source is column, target is row. **(B)** Mean PLS-PC matrix (left: AD group, right: HC group). Color range is [–1, 1] **(C)** Mean pwGC matrix (left: AD group, right: HC group). Color range is [–3σ, 3σ] **(D)** Mean VARDNN-DI matrix (left: AD group, right: HC group). Color range is [–3σ, 3σ] **(E)** Cosine similarity of the mean matrix between the AD and HC group matrices. **(F)** Lilliefors test result matrix of FC (left) and VARDNN-DI (right) of HC.

FIGURE 9 | (A) Significantly different ($p < 0.05$, yellow color; non-significant, blue color) regional relationship matrices between AD and HC groups. From left to right, FC, PLS-PC, pwGC, and VARDNN-DI, respectively. **(B)** The whole brain image of the most different ($0 \leq p < 0.0024$) regional relationships by VARDNN-DI. The image on the left is seen from the dorsal view and the right image shows the medial view of the left hemisphere. Posterior is left and anterior is right. Blue circles are region centroids. The source to target areas are color coded from green to red. A smaller p-value shows a thicker line.

Furthermore, functional connectome estimation from ground-truth synthetic neural activity signals generated by the reduced Wong–Wang model was evaluated. VARDNN-based dFC measures significantly outperformed other causal estimation measures for the large node count case (**Figure 6**). Several articles evaluating undirected or directed functional connectivity algorithms based on synthetic fMRI BOLD signals generated by DCM exist, for example, Smith et al. (2011) and Prando et al. (2020). However, the results of these experiments are difficult to compare due to their various network configurations and densities. For example, networks with node numbers of 5, 10, and 50 with very sparse connections were used in Smith et al. (2011), while node numbers of 7 and 66, with a network density 0.5 were used in Prando et al. (2020). In this work we evaluated both increasing network density and node number cases. The evaluation of multiple types of algorithms for the early diagnosis of Alzheimer's disease has been performed by Dauwels et al. (2010). We designed a subject-wise evaluation framework to separate AD and HC groups from empirical rsfMRI BOLD data in order to quantify the performance of functional connectome estimation measures. In our results, VARDNN-DI especially succeeded in extracting lesioned brain regions consistent with previous studies (**Table 3**), and it showed a higher AUC score over existing functional connectome estimation measures when separating AD and HC groups (**Figure 10**). This result also supports the sensitivity of the VARDNN-DI measure.

Both multivariate Granger causality and LINUE transfer entropy rely on linear multivariate auto-regression. Thus,

their sensitivity becomes low for large node counts and little sample data (Barnett and Seth, 2014). Several countermeasure techniques, such as PCA (Zhou et al., 2009), Elastic Net (Zou and Hastie, 2005; Ryali et al., 2012), and PLS (Wold et al., 2001; Krishnan et al., 2011) have been used to avoid these large dimensionality and multi-collinearity problems. We compared VARDNN against these techniques and VARDNN showed competitive or even better estimation results. If a DNN does not have (non-linear) activation functions, it is essentially the same as linear regression. Activation functions, such as ReLU, non-linearly extend the ability of linear regression. This strategy works well for extending the capability of GC by VARDNN. The well-studied training techniques of deep neural networks complements the weaknesses of linear regression and extends its capabilities.

We showed that VARDNN-based GC was able to obtain better results than RNN-GC, an established neural network-based approach. RNN-GC was directly compared with another neural network-based approach called NN-GC, which it outperformed (Montalto et al., 2015; Wang et al., 2018). The original NN-GC definition of causality was a simple subtraction: $Err_{reduced} - Err_{full}$, and this definition showed less sensitivity than the RNN-GC definition of $\log\left(var\left(Err_{reduced}\right)/var\left(Err_{full}\right)\right)$. Our VARDNN-GC measure used the same type of definition as RNN-GC and outperformed it. From the comparative results of Wang et al. (2018) we can also expect VARDNN to outperform NN-GC. RNNs require sequential input, which can make training convergence difficult. Our approach uses an auto-regressive DNN

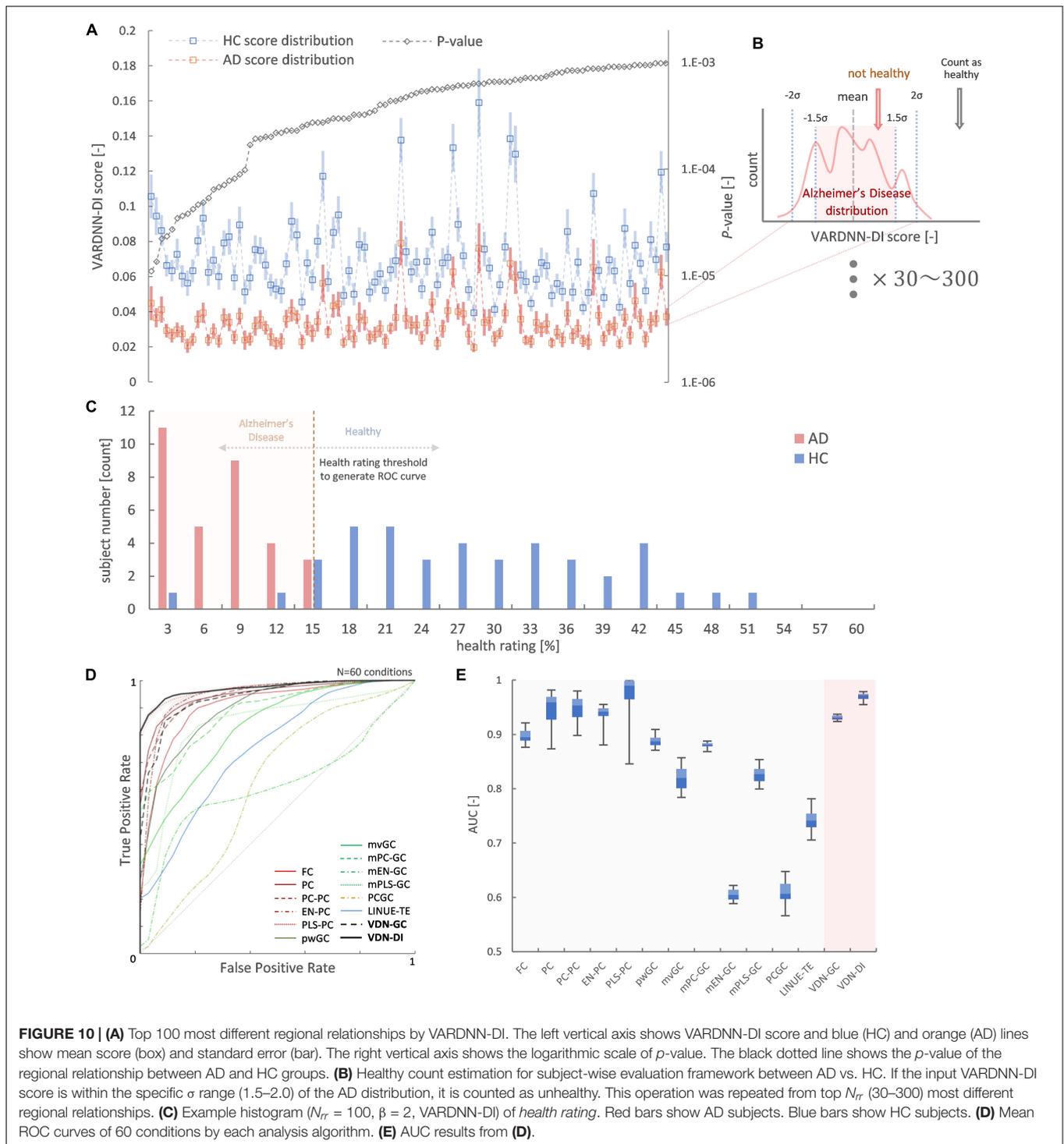**TABLE 3 |** Top 20 regions of most different dFC between AD and HC, calculated from VARDNN-DI.

ROI name	Network	FC	PC-PC	PLS-PC	pGCSrc	pGCTrg	VDNdiSrc	VDNdiTrg	VDNdiTotal	References
Precuneous (Precuneous cortex)	pDMN	20	14	8	11	18	30	59	**89**	Bakkour et al., 2013; Peraza et al., 2014; Utevsky et al., 2014; Zott et al., 2018; Pascoal et al., 2019
pSMG l (Supramarginal gyrus, posterior division left)	FPN/Lang (L)	24	13	5	12	4	37	43	**80**	Harasty et al., 1999; Bakkour et al., 2013; Peraza et al., 2014
ICC l (Intracalcarine cortex left)	Visual.Primary	6	12	9	10	4	19	58	**77**	Peraza et al., 2014; Utevsky et al., 2014; Hafkemeijer et al., 2015; Schwab et al., 2020
SCC l (Supracalcarine cortex left)	Visual.Primary	14	8	13	10	12	10	66	**76**	Utevsky et al., 2014; Hafkemeijer et al., 2015
AG l (Angular gyrus left)	lDMN	13	16	6	8	26	18	58	**76**	Bakkour et al., 2013
PC (Cingulate gyrus, posterior division)	pDMN	5	9	8	19	10	35	24	**59**	Bakkour et al., 2013
toITG l (Inferior temporal gyrus, temporooccipital part left)	DAN	17	9	6	8	1	18	41	**59**	Mosconi et al., 2009; Bakkour et al., 2013; Peraza et al., 2014; Utevsky et al., 2014; Zott et al., 2018; Pascoal et al., 2019
CO l (Central opercular cortex left)	Auditory	9	9	5	6	7	35	23	**58**	
Cuneal r (Cuneal cortex right)	Visual.Primary	6	9	7	9	3	14	41	**55**	Utevsky et al., 2014; Hafkemeijer et al., 2015
Cereb6 l (Cerebelum 6 left)	Cerebellum	10	2	7	15	15	33	19	**52**	Jacobs et al., 2018
SFG r (Superior frontal gyrus right)	FPN (R)	22	15	12	12	11	37	14	**51**	Bakkour et al., 2013
Pallidum l	Salience	13	9	7	16	8	18	32	**50**	Bakkour et al., 2013
SPL r (Superior parietal lobule right)	DAN	36	12	5	12	6	34	16	**50**	Peraza et al., 2014
Cereb9 l (Cerebelum 9 left)	Cerebellum/pPaHC	10	11	7	7	8	42	7	**49**	Jacobs et al., 2018
FOrb l (Frontal orbital cortex left)	Inferior Temporal	3	7	4	4	13	37	12	**49**	
Caudate r	Thalamus	16	6	8	5	35	28	20	**48**	
AG r (Angular gyrus right)	lDMN	15	3	9	10	0	23	24	**47**	
Cereb45 l (Cerebelum 4 5 Left)	Cerebellum	9	10	5	12	22	16	31	**47**	Peraza et al., 2014
sLOC l (Lateral occipital cortex, superior division Left)	sLOC	17	4	5	4	19	28	19	**47**	Bakkour et al., 2013
Ver12 (Vermis 1 2)	Cerebellum/pPaHC	34	9	6	7	3	19	27	**46**	

From left to right column: ROI name, brain network name, significantly different region counted by FC, PC-PC, PLS-PC, pwGC source count, pwGC target count, VARDNN-DI source count, VARDNN-DI target count, total of VDNdiSrc and VDNdiTrg column, references, respectively.

structure so mini-batch and shuffling during network training becomes possible. This flexibility improved the training time of the VARDNN and the accuracy of causality estimation. Furthermore, we defined a new dFC measure, VARDNN-DI, as $\left| z_{full} - z_{reduced} \right|$, to extract causal relationships from trained DNN weights. This simple equation showed surprisingly high sensitivity in comparison to other algorithms. Owing to these properties, the VARDNN can be trained on non-linear signals and capture non-linear relationships in the weights of the DNNs.

Despite its effectiveness, there are some limitations to the VARDNN and its functional connectome measurements. Since the designs of the VARDNN-based dFC measures have similarities with *predictive*-based analysis, they have similar limitations to Granger causality (Bressler and Seth, 2011;

Barnett and Seth, 2014; Stokes and Purdon, 2017). The first is variability in the shape and latency of Hemodynamic Response Functions (HRFs) in different brain regions (Miezin et al., 2000) and across subjects (Aguirre et al., 1998). Neural activity and HRFs in brain regions are not completely synchronized, so data driven functional connectome measures, including FC, PC, and GC, have always faced this issue. However, many biological experiments involve two group comparisons, for example AD vs. HC, wild type vs. mutated disease model, etc. In this scheme, the variability of HRFs in different brain regions and subjects can be considered as differentiating features between the two groups. Therefore, for such schemes, applying data driven functional connectome measures for rsfMRI BOLD signal analysis and striving to increasing the experimental number will support

FIGURE 10 | (A) Top 100 most different regional relationships by VARDNN-DI. The left vertical axis shows VARDNN-DI score and blue (HC) and orange (AD) lines show mean score (box) and standard error (bar). The right vertical axis shows the logarithmic scale of *p*-value. The black dotted line shows the *p*-value of the regional relationship between AD and HC groups. **(B)** Healthy count estimation for subject-wise evaluation framework between AD vs. HC. If the input VARDNN-DI score is within the specific σ range (1.5–2.0) of the AD distribution, it is counted as unhealthy. This operation was repeated from top N_{rr} (30–300) most different regional relationships. **(C)** Example histogram ($N_{rr} = 100$, β = 2, VARDNN-DI) of *health rating*. Red bars show AD subjects. Blue bars show HC subjects. **(D)** Mean ROC curves of 60 conditions by each analysis algorithm. **(E)** AUC results from **(D)**.

robustness of experimental results. The second issue is related to changes in the sampling time interval. In our experiments, VARDNN-DI detected lesioned brain regions from experimental rsfMRI BOLD signals with a repetition time (TR) of 3 s. These dFCs clearly depend on the repetition time. If the repetition time was changed to 1.5 s, the dFC results might have differed. This problem is not only for *predictive*-based analysis, but also for

zero-lag analysis. Longer time intervals average over fine patterns in neural activity, and correlation-based analysis shows better sensitivity at lower as compared to higher sampling rates. Thus, we need to be careful and consider this time dependence when choosing the appropriate analysis algorithm. The third is with regard to negative causal relationships between nodes. The DCM inversion algorithm is able to estimate positive and negative

connection weights, but GC-based algorithms basically do not have this function because they are calculated from changes in variance. Some GC approaches may generate negative values, but such values have no physical interpretation (Chen et al., 2006). This type of positive and negative causality estimation is also an issue for VARDNN.

We provide a VARDNN toolbox for MATLAB that implements the algorithms described in this paper. The code is open source and can be downloaded from the GitHub repository at https://github.com/takuto-okuno-riken/vardnn. The VARDNN toolbox contains a command line tool that includes several analysis algorithms, namely VARDNN-DI, VARDNN-GC, mvGC, pwGC, linear TE, FC, and PC to estimate a functional connectome from brain signals. We hope that this toolbox will help further brain analysis, and the toolbox will be expanded with new features in the future. The Deep Learning toolbox ver. 12.1 (or later), Fuzzy Logic Toolbox ver. 2.6 (or later), and MATLAB R2019a (or later) are required to run the VARDNN scripts.

DATA AVAILABILITY STATEMENT

The original contributions presented in the study are included in the article/**Supplementary Material**, further inquiries can be directed to the corresponding author/s.

ETHICS STATEMENT

The studies involving human participants were reviewed and approved by the Alzheimer's Disease Neuroimaging Initiative. The patients/participants provided their written informed consent to participate in this study.

AUTHOR CONTRIBUTIONS

TO contributed to conception and design of the study and wrote the first draft of the manuscript. Both authors contributed to manuscript revision, read, and approved the submitted version.

ACKNOWLEDGMENTS

SUPPLEMENTARY MATERIAL

The Supplementary Material for this article can be found online at: https://www.frontiersin.org/articles/10.3389/fnins.2021.764796/full#supplementary-material

REFERENCES

Aguirre, G. K., Zarahn, E., and D'Esposito, M. (1998). The variability of human, bold hemodynamic responses. *Neuroimage* 8, 360–369.

Alstott, J., Breakspear, M., Hagmann, P., Cammoun, L., and Sporns, O. (2009). Modeling the impact of lesions in the human brain. *PLoS Comput. Biol.* 5:e1000408. doi: 10.1371/journal.pcbi.1000408

Bakkour, A., Morris, J. C., Wolk, D. A., and Dickerson, B. C. (2013). The effects of aging and Alzheimer's disease on cerebral cortical anatomy: specificity and differential relationships with cognition. *Neuroimage* 76, 332–344. doi: 10.1016/j.neuroimage.2013.02.059

Barnett, L., Barrett, A. B., and Seth, A. K. (2009). Granger causality and transfer entropy are equivalent for Gaussian variables. *Phys. Rev. Lett.* 103:238701.

Barnett, L., and Seth, A. K. (2014). The MVGC multivariate granger causality toolbox: a new approach to granger-causal inference. *J. Neurosci. Methods* 223, 50–68. doi: 10.1016/j.jneumeth.2013.10.018

Bressler, S. L., and Seth, A. K. (2011). Wiener–granger causality: a well established methodology. *Neuroimage* 58, 323–329. doi: 10.1016/j.neuroimage.2010.02.059

Chen, C., Tyler, C. W., and Baseler, H. A. (2003). Statistical properties of bold magnetic resonance activity in the human brain. *Neuroimage* 20, 1096–1109.

Chen, Y., Bressler, S. L., and Ding, M. (2006). Frequency decomposition of conditional Granger causality and application to multivariate neural field potential data. *J. Neurosci. Methods* 150, 228–237. doi: 10.1016/j.jneumeth.2005.06.011

Chivukula, A. S., Li, J., and Liu, W. (2018). "Discovering Granger-causal features from deep learning networks," in *AI 2018: Advances in Artificial Intelligence*, Vol. 11320, eds T. Mitrovic, B. Xue, and X. Li (Cham: Springer), 692–705.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Math. Control Signal. Syst.* 2, 303–314. doi: 10.1007/bf02551274

Dauwels, J., Vialatte, F., Musha, T., and Cichocki, A. (2010). A comparative study of synchrony measures for the early diagnosis of Alzheimer's disease based on EEG. *Neuroimage* 49, 668–693. doi: 10.1016/j.neuroimage.2009.06.056

Dauwels, J., Vialatte, F., Rutkowski, T., and Cichocki, A. (2008). Measuring neural synchrony by message passing. *Adv. Neural Inform. Process. Syst.* 20, 361–368.

Deco, G., Ponce-Alvarez, A., Mantini, D., Romani, G. L., Hagmann, P., and Corbetta, M. (2013). Resting-state functional connectivity emerges from structurally and dynamically shaped slow linear fluctuations. *J. Neurosci.* 33, 11239–11252. doi: 10.1523/JNEUROSCI.1091-13.2013

Delalleau, O. and Bengio, Y. (2011). Shallow vs. deep sum-product networks. *Adv. Neural Inf. Process. Syst.* 24, 666–674.

Duggento, A., Guerrisi, M., and Toschi, N. (2019). Echo state network models for nonlinear Granger causality. *bioRxiv*[Preprint] bioRxiv: 651679,

Friston, K., Harrison, L., and Penny, W. (2003). Dynamic causal modelling. *Neuroimage* 19, 1273–1302.

Friston, K., Moran, R., and Seth, A. K. (2013). Analysing connectivity with Granger causality and dynamic causal modelling. *Curr. Opin. Neurobiol.* 23, 172–178.

Friston, K. J., Frith, C. D., Liddle, P. F., and Sj Franckowiak, R. (1993). Functional connectivity: the principal-component analysis of large (PET) data sets. *J. Cereb. Blood Flow Metab.* 13, 5–14. doi: 10.1038/jcbfm.1993.4

Gămănuţ, R., Kennedy, H., Toroczkai, Z., Ercsey-Ravasz, M., Van Essen, D. C., Knoblauch, K., et al. (2018). The mouse cortical connectome, characterized by an ultra-dense cortical graph, maintains specificity by distinct connectivity profiles. *Neuron* 97, 698–715.e10. doi: 10.1016/j.neuron.2017.12.037

Gao, Q., Duan, X., and Chen, H. (2011). Evaluation of effective connectivity of motor areas during motor imagery andexecution using conditional Granger causality. *Neuroimage* 54, 1280–1288. doi: 10.1016/j.neuroimage.2010.08.071

Gerstein, G. L., Bedenbaugh, P., and Aertsen, A. M. H. J. (1989). Neuronal assemblies. *IEEE Trans. Biomed. Eng.* 36, 4–14.

Geweke, J. (1982). Measurement of linear dependence and feedback between multiple time series. *J. Am. Stat. Assoc.* 77, 304–313. doi: 10.1080/01621459.1982.10477803

Glorot, X., Bordes, A., and Bengio, Y. (2011). "Deep sparse rectifier neural networks," *Proceedings of the 14th International Conference on Artificial Intelligence and Statistics*, Vol. 15 (Montreal, QC: Department of IRO University of Montreal). 315–323.

Goebel, R., Roebroeck, A., Kim, D. S., and Formisano, E. (2003). Investigating directed cortical interactions in time-resolved fMRI data using vector autoregressive modeling and Granger causality mapping. *Magn. Reson. Imaging* 21, 1251–1261. doi: 10.1016/j.mri.2003.08.026

Goodfellow, I., Bengio, Y., and Courville, A. (2016). *Deep Learning*. Cambridge, MA: MIT Press.

Goodfellow, I. J., Bulatov, Y., Ibarz, J., Arnoud, S., and Shet, V. (2014). Multi-digit number recognition from street viewimagery using deep convolutional neural networks. *arXiv*[Preprint] arXiv: 1312.6082.

Granger, C. (1969). Investigating causal relations by econometric models and cross-spectral methods. *Econometrica* 37, 424–438. doi: 10.2307/1912791

Hafkemeijer, A., Möller, C., Dopper, E. G., Jiskoot, L. C., Schouten, T. M., van Swieten, J. C., et al. (2015). Resting state functional connectivity differences between behavioral variant frontotemporal dementia and Alzheimer's disease. *Front. Hum. Neurosci.* 9:474. doi: 10.3389/fnhum.2015.00474

Hahn, G., Skeide, M. A., Mantini, D., Ganzetti, M., Destexhe, A., Friederici, A. D., et al. (2019). A new computational approach to estimate whole-brain effective connectivity from functional and structural MRI, applied to language development. *Sci. Rep.* 9:8479. doi: 10.1038/s41598-019-44909-6

Harasty, J. A., Halliday, G. M., Kril, J. J., and Code, C. (1999). Specific temporoparietal gyral atrophy reflects thepattern of language dissolution in Alzheimer's disease. *Brain* 122, 675–686. doi: 10.1093/brain/122.4.675

Honey, C. J., Kotter, R., Breakspear, M., and Sporns, O. (2007). Network structure of cerebral cortex shapes functional connectivity on multiple time scales. *Proc. Natl Acad. Sci. U. S. A.* 104, 10240–10245. doi: 10.1073/pnas.0701519104

Hornik, K. (1989). Multilayer feedforward networks are universal approximators. *Neural Netw.* 2, 359–366.

Jacobs, H. I. L., Hopkins, D. A., Mayrhofer, H. C., Bruner, E., van Leeuwen, F. W., Raaijmakers, W., et al. (2018). The cerebellum in Alzheimer's disease: evaluating its role in cognitive decline. *Brain* 141, 37–47. doi: 10.1093/brain/awx194

Kingma, D. P., and Ba, J. (2014). Adam: a Method for Stochastic Optimization. *arXiv*[Preprint] arXiv: 1412.6980,

Krishnan, A., Williams, L. J., McIntosh, A. R., and Abdi, H. (2011). Partial Least Squares (PLS) methods for neuroimaging: a tutorial and review. *Neuroimage* 56, 455–475. doi: 10.1016/j.neuroimage.2010.07.034

Krueger, F., Landgraf, S., Meer, E., Deshpande, G., and Hu, X. (2011). Effective connectivity of the multiplication network: a functional MRI and multivariate granger causality mapping study. *Hum. Brain Mapp.* 32, 1419–1431. doi: 10.1002/hbm.21119

Li, X., Morgan, P. S., Ashburner, J., Smith, J., and Rorden, C. (2016). The first step for neuroimaging data analysis: DICOM to NIfTI conversion. *J. Neurosci. Methods* 264, 47–56. doi: 10.1016/j.jneumeth.2016.03.001

Liang, S., and Srikant, R. (2016). Why deep neural networks for function approximation? *arXiv*[Preprint] arXiv: 1610.04161,

Marinazzo, D., Liao, W., Chen, H., and Stramaglia, S. (2011). Nonlinear connectivity by Granger causality. *Neuroimage* 58, 330–338. doi: 10.1016/j.neuroimage.2010.01.099

Marinazzo, D., Pellicoro, M., and Stramaglia, S. (2012). Causal information approach to partial conditioning in multivariate data sets. *Comp. Math. Methods. Med.* 2012:303601. doi: 10.1155/2012/303601

Matsumoto, M., and Nishimura, T. (1998). A 623-dimensionally equidistributed uniform pseudorandom number generator. *ACM Trans. Model. Comput. Simul.* 8, 3–30.

Melozzi, F., Woodman, M. M., Jirsa, V. K., and Bernard, C. (2017). The virtual mouse brain: a computational neuroinformatics platform to study whole mouse brain dynamics. *eNeuro* 4:ENEURO.0111-17.2017., doi: 10.1523/ENEURO.0111-17.2017

Menon, V., and Uddin, L. Q. (2010). Saliency, switching, attention and control: a network model of insula function. *Brain Struct. Funct.* 214, 655–667. doi: 10.1007/s00429-010-0262-0

Miezin, F. M., Maccotta, L., Ollinger, J. M., Petersen, S. E., and Buckner, R. L. (2000). Characterizing the hemodynamic response: effects of presentation rate, sampling procedure, and the possibility of ordering brain activity based on relative timing. *Neuroimage* 11, 735–759. doi: 10.1006/nimg.2000.0568

Montalto, A., Faes, L., and Marinazzo, D. (2014). MuTE: a MATLAB toolbox to compare established and novel estimators of the multivariate transfer entropy. *PLoS One* 9:e109462. doi: 10.1371/journal.pone.0109462

Montalto, A., Stramaglia, S., Faes, L., Tessitore, G., Prevete, R., and Marinazzo, D. (2015). Neural networks with non-uniform embedding and explicit validation phase to assess granger causality. *Neural Netw.* 71, 159–171. doi: 10.1016/j.neunet.2015.08.003

Mosconi, L., Pupi, A., and De Leon, M. J. (2009). Brain glucose hypometabolism and oxidative stress in preclinical Alzheimer's disease. *Ann. N. Y. Acad. Sci.* 1147, 180–195.

Nair, V., and Hinton, G. E. (2010). "Rectified linear units improve restricted Boltzmann machines," in *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, Haifa, 807–814.

Oh, S., Harris, J. A., Ng, L., Winslow, B., Cain, N., Mihalas, S., et al. (2014). A mesoscale connectome of the mouse brain. *Nature* 508, 207–214.

Pascoal, T. A., Mathotaarachchi, S., Kang, M. S., Mohaddes, S., Shin, M., and Park, A. Y. (2019). Aβ-induced vulnerability propagates via the brain's default mode network. *Nat. Commun.* 10:2353. doi: 10.1038/s41467-019-10217-w

Peraza, R. P., Kaiser, M., Firbank, M., Graziadio, S., Bonanni, L., and Onofrj, M. (2014). fMRI resting state networks and their association with cognitive fluctuations in dementia with Lewy bodies. *Neuroimage* 4, 558–565. doi: 10.1016/j.nicl.2014.03.013

Prando, G., Zorzi, M., Bertoldo, A., Corbetta, M., Zorzi, M., and Chiuso, A. (2020). Sparse DCM for whole-brain effective connectivity from resting-state fMRI data. *Neuroimage* 208:116367. doi: 10.1016/j.neuroimage.2019.116367

Ryali, S., Chen, T., Supekar, K., and Menon, V. (2012). Estimation of functional connectivity in fMRI data using stability selection-based sparse partial correlation with elastic net penalty. *Neuroimage* 59, 3852–3861. doi: 10.1016/j.neuroimage.2011.11.054

Schreiber, T. (2000). Measuring information transfer. *Phys. Rev. Lett.* 85:461. doi: 10.1103/physrevlett.85.461

Schwab, S., Afyouni, S., Chen, Y., Han, Z., Guo, Q., and Dierks, T. (2020). Functional connectivity alterations of the temporal lobe and hippocampus in semantic dementia and Alzheimer's disease. *J. Alzheimers Dis.* 76, 1461–1475.

Shannon, C. (1948). Amathematical theory of communication. *Bell Syst. Tech. J.* 27, 379–423.

Shimizu, S., Hoyer, P. O., Hyvarinen, A., and Kerminen, A. (2006). A Linear Non-Gaussian Acyclic Model for Causal Discovery. *J. Mach. Learn. Res.* 7, 2003–2030.

Smith, S. M., Miller, K. L., Salimi-Khorshidi, G., Webster, M., Beckmann, C. F., Nichols, T. E., et al. (2011). Network modelling methods for FMRI. *Neuroimage* 54, 875–891. doi: 10.1016/j.neuroimage.2010.08.063

Stokes, P. A., and Purdon, P. L. (2017). A study of problems encountered in Granger causality analysis from a neuroscience perspective. *Proc. Natl. Acad. Sci. U. S. A.* 114, E7063–E7072.

Utevsky, A. V., Smith, D. V., and Huettel, S. A. (2014). Precuneus is a functional core of the default-mode network. *J. Neurosci.* 34, 932–940. doi: 10.1523/jneurosci.4227-13.2014

Valdes-Sosa, P. A., Roebroeck, A., Daunizeau, J., and Friston, K. (2011). Effective connectivity: influence, causality and biophysical modeling. *Neuroimage* 58, 339–361. doi: 10.1016/j.neuroimage.2011.03.058

Wang, Y., Lin, K., Qi, Y., Lian, Q., Feng, S., Wu, Z., et al. (2018). Estimating brain connectivity with varying-length time lags using a recurrent neural network. *IEEE Trans. Biomed. Eng* 65, 1953–1963. doi: 10.1109/TBME.2018.2842769

Wen, X., Yao, L., Liu, Y., and Ding, M. (2012). Causal interactions in attention networks predict behavioral performance. *J. Neurosci.* 32, 1284–1292. doi: 10.1523/jneurosci.2817-11.2012

Whitfield-Gabrieli, S., and Nieto-Castanon, A. (2012). Conn: a functional connectivity toolbox for correlated and anticorrelated brain networks. *Brain Connect.* 2, 125–141. doi: 10.1089/brain.2012.0073

Wold, S., Sjöström, M., and Eriksson, L. (2001). PLS-regression: a basic tool of chemometrics. *Chem. Int. Lab. Syst.* 58, 109–130. doi: 10.1016/s0169-7439(01)00155-1

Wong, K. F., and Wang, X. J. (2006). A recurrent network mechanism of time integration in perceptual decisions. *J. Neurosci.* 26, 1314–1328. doi: 10.1523/jneurosci.3733-05.2006

Zhou, Z., Chen, Y., Ding, M., Wright, P., Lu, Z., and Liu, Y. (2009). Analyzing brain networks with PCA and conditional granger causality. *Hum. Brain Mapp.* 30, 2197–2206. doi: 10.1002/hbm.20661

Zott, B., Busche, M. A., Sperling, R. A., and Konnerth, A. (2018). What happens with the circuit in Alzheimer's disease in mice and humans?". *Annu. Rev. Neurosci.* 41, 277–297. doi: 10.1146/annurev-neuro-080317-061725

Zou, H., and Hastie, T. (2005). Regularization and variable selection via the elastic net. *J. R. Stat. Soc. B* 67(Pt 2), 301–320. doi: 10.1111/j.1467-9868.2005.00503.x