Semi-supervised Tensor Factorization for Brain Network Analysis

Bokai Cao[1], Chun-Ta Lu[1], Xiaokai Wei[1], Philip S. Yu[1,2], and Alex D. Leow[3]

[1]Department of Computer Science, University of Illinois, Chicago, IL, USA
[2]Institute for Data Science, Tsinghua University, Beijing, China
[3]Department of Psychiatry, University of Illinois, Chicago, IL, USA
{caobokai,clu29,xwei2,psyu}@uic.edu, alexfeuillet@gmail.com

Abstract. Brain networks characterize the temporal and/or spectral connections between brain regions and are inherently represented by multi-way arrays (tensors). In order to discover the underlying factors driving such connections, we need to derive compact representations from brain network data. Such representations should be discriminative so as to facilitate the identification of subjects performing different cognitive tasks or with different neurological disorders. In this paper, we propose semiBAT, a novel semi-supervised Brain network Analysis approach based on constrained Tensor factorization. semiBAT 1) leverages unlabeled resting-state brain networks for task recognition, 2) explores the temporal dimension to capture the progress, 3) incorporates classifier learning procedure to introduce supervision from labeled data, and 4) selects discriminative latent factors for different tasks. The Alternating Direction Method of Multipliers (ADMM) framework is utilized to solve the optimization objective. Experimental results on EEG brain networks illustrate the superior performance of the proposed semiBAT model on graph classification with a significant improvement 31.60% over plain vanilla tensor factorization. Moreover, the data-driven factors can be readily visualized which should be informative for investigating cognitive mechanisms.

Keywords: brain network, graph mining, tensor factorization

1 Introduction

Brain networks (*a.k.a*, connectome [24]) obtained from neuroimaging data have been commonly employed to study neuropsychiatric disorders [3,15,27,30]. Connectivity patterns are usually embedded within the graph structures by a set of vertices and edges where vertices correspond to regions of interest in the brain and edges represent the connectivity strength or correlation between brain regions. Considering the temporal and spectral domain, original brain networks are typically represented in multi-way arrays (*i.e.*, tensors) which make the conventional vector-based classification algorithms inapplicable. Moreover, directly reshaping tensors into vectors would result in the curse of dimensionality, and

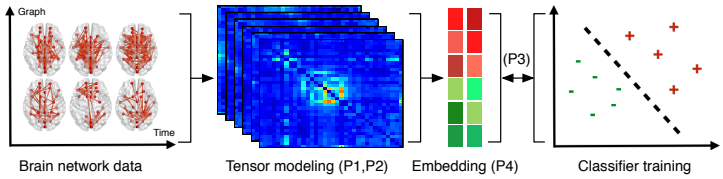


Fig. 1. The framework of tensor-based brain network analysis.

the number of brain network samples is usually small, thereby making it challenging to train an effective classifier in a high dimensional feature space with a limited number of samples.

In order to apply conventional machine learning algorithms and train pattern classifiers, it is preferable to first derive vector representations from the brain network data. In general, researchers have proposed to extract two types of features: (1) graph-theoretical measures [27,13] and (2) subgraph patterns [17,6]. However, the expressiveness of these features is limited to the predefined formulations. To explore a larger space of potentially informative features to represent brain networks, it motivates us to learn latent representations from the brain network data. It is desirable to let the latent representations be discriminative so that brain networks with different labels can easily be separated. Learning such representations is a non-trivial task due to the following problems:

(P1) Although labeled brain network data for specific tasks or diseases are usually costly to obtain, brain networks under resting-state from healthy subjects are recorded in many neuroimaging experiments. How can we leverage the unlabeled data, *i.e.*, resting-state brain networks, to facilitate classification?
(P2) Existing studies usually compute time-averaged brain networks before further analysis [28] which may result in formidable information loss. How can we directly fully utilize the temporal information in our model?
(P3) In order to obtain discriminative representations, we should incorporate the classifier training procedure into the representation learning process for leveraging the supervision information. How can we effectively fuse these two procedures together?
(P4) Different classes (or tasks) are usually associated with different subsets of the latent factors. How can we achieve feature selection in the latent space?

In this paper, we propose SEMIBAT, a semi-supervised Brain network Analysis approach based on constrained Tensor factorization. The proposed framework is illustrated in Figure 1. The contributions of this work are fourfold:

- We leverage unlabeled resting-state brain networks together with labeled data to collectively learn a latent space, which alleviates the problem that labeled brain network data for specific tasks or diseases are usually very limited.
- We model brain networks through partially symmetric tensor factorization which is suitable for inherently undirected graphs, *e.g.*, EEG brain networks. The temporal dimension is modeled as one of modes in the fourth-order tensor.

Table 1. Overview of tensor notation and operators.

Notation	Interpretation
a	scalar
$\mathbf{a}$	vector
$\mathbf{A}$	matrix
$\mathcal{X}$	tensor, set or space
$\mathbf{X}_{(k)}$	matricization of tensor $\mathcal{X}$ along mode k
$*$	Hadamard product (elementwise product)
$\circ$	tensor product (outer product)
$\times_k$	mode-k product
$\otimes$	Kronecker product
$\odot$	Khatri-Rao product
$\|\cdot\|$	norm of a vector, matrix or tensor
$\|\cdot\|$	cardinality of a set

– We blend representation learning and classifier training into a unified optimization problem, which allows classifier parameters to interact with discriminative latent factors by leveraging the supervision information.
– We incorporate the $\ell_{2,1}$ norm to conduct feature selection in the latent space, thereby identifying discriminative latent factors for different tasks.

2 Preliminaries

Table 1 lists some basic symbols that will be used throughout the paper. We introduce the concept of tensors which are higher order arrays that generalize the notions of vectors (the first-order tensors) and matrices (the second-order tensors), whose elements are indexed by more than two indices. Each index expresses a mode of the data and corresponds to a coordinate direction. The number of variables in each mode indicates the dimensionality of a mode. The order of a tensor is determined by the number of its modes. An mth-order tensor can be represented as $\mathcal{X} = (x_{i_1,\cdots,i_m}) \in \mathbb{R}^{I_1 \times \cdots \times I_m}$, where I_i is the dimension of $\mathcal{X}$ along mode i. An overview of tensor notation and operators is given as follows which will be used to formulate the problem.

Definition 1 (Tensor Product). *The tensor product $\mathcal{X} \circ \mathcal{Y}$ of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_m}$ and another tensor $\mathcal{Y} \in \mathbb{R}^{I'_1 \times \cdots \times I'_{m'}}$ is defined by $(\mathcal{X} \circ \mathcal{Y})_{i_1,\ldots,i_m,i'_1,\ldots,i'_{m'}} = x_{i_1,\ldots,i_m} y_{i'_1,\ldots,i'_{m'}}$.*

Tensor product is also referred to as outer product in some literature [9,29]. An mth-order tensor is a rank-one tensor if it can be defined as the tensor product of m vectors.

Definition 2 (Mode-k Product). *The mode-k product $\mathcal{X} \times_k \mathbf{A}$ of a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_m}$ and a matrix $\mathbf{A} \in \mathbb{R}^{J \times I_k}$ is of size $I_1 \times \cdots \times I_{k-1} \times J \times I_{k+1} \times \cdots \times I_m$ and is defined by $(\mathcal{X} \times_k \mathbf{A})_{i_1,\ldots,i_{k-1},j,i_{k+1},\ldots,i_m} = \sum_{i_k=1}^{I_k} x_{i_1,\ldots,i_m} a_{j,i_k}$.*

Definition 3 (Kronecker Product). *The Kronecker product of two matrices* $\mathbf{A} \in \mathbb{R}^{I \times J}, \mathbf{B} \in \mathbb{R}^{K \times L}$ *is of size* $IK \times JL$ *and is defined by*

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & \cdots & a_{1J}\mathbf{B} \\ \vdots & \ddots & \vdots \\ a_{I1}\mathbf{B} & \cdots & a_{IJ}\mathbf{B} \end{pmatrix}$$

Definition 4 (Khatri-Rao Product). *The Khatri-Rao product of two matrices* $\mathbf{A} \in \mathbb{R}^{I \times K}, \mathbf{B} \in \mathbb{R}^{J \times K}$ *is of size* $IJ \times K$ *and is defined by* $\mathbf{A} \odot \mathbf{B} = (a_1 \otimes b_1, \cdots, a_K \otimes b_K)$ *where* $a_1, \cdots, a_K, b_1, \cdots, b_K$ *are the columns of matrices* $\mathbf{A}$ *and* $\mathbf{B}$*, respectively.*

Definition 5 (Partially Symmetric Tensor). *A rank-one* m*th-order tensor* $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_m}$ *is partially symmetric if it is symmetric on modes* $i_1, ..., i_j \in \{1, ..., m\}$ *and can be written as the tensor product of* m *vectors:* $\mathcal{X} = \mathbf{x}^{(1)} \circ \cdots \circ \mathbf{x}^{(m)}$ *where* $\mathbf{x}^{(i_1)} = \cdots = \mathbf{x}^{(i_j)}$.

Definition 6 (Mode-k Matricization). *The mode-k matricization of a tensor* $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_m}$ *is denoted by* $\mathbf{X}_{(k)}$ *and arranges the mode-k fibers to be the columns of the resulting matrix. The dimension of* $\mathbf{X}_{(k)}$ *is* $\mathbb{R}^{I_k \times J}$*, where* $J = I_1 \cdots I_{k-1} I_{k+1} \cdots I_m$*. Each tensor element* $(i_1, \cdots, i_m)$ *maps to the matrix element* (i_k, j)*:* $j = 1 + \sum_{p=1, p \neq k}^{m} (i_p - 1) J_p$ *with* $J_p = \prod_{q=1, q \neq k}^{p-1} I_q$.

3 SEMIBAT Framework

3.1 Problem Formulation

Let $\mathcal{D} = \{G_1, \cdots, G_n\}$ denote a dynamic graph dataset of brain networks where $|\mathcal{D}| = n$ is the number of graph objects. All graphs in the dataset share a given set of vertices V which corresponds to a brain parcellation scheme. Suppose the brain is parcellated via an atlas into $|V| = m$ regions, and the temporal dimensionality is t. A brain network G_i can be represented by a partially symmetric tensor $\mathcal{Z}_i \in \mathbb{R}^{m \times m \times t}$. We assume that the first l graphs within $\mathcal{D}$ are labeled and $\mathbf{Y} \in \mathbb{R}^{l \times c}$ is the class label matrix where c is the number of class labels. $\mathbf{Y}(i, j) = 1$ if G_i belongs to the j-th class, otherwise $\mathbf{Y}(i, j) = 0$. For convenience, we also denote the labeled graph dataset by $\mathcal{D}_l = \{G_1, \cdots, G_l\}$, and the unlabeled graph dataset as $\mathcal{D}_u = \{G_{l+1}, \cdots, G_n\}$, $\mathcal{D} = \mathcal{D}_l \cup \mathcal{D}_u$. In our experiments, brain networks under emotion regulation tasks compose labeled graphs, while those under resting-state compose unlabeled graphs.

3.2 Tensor Modeling

We first address the problem (P1) discussed in Section 1 by stacking the brain network dataset $\mathcal{D}$ of n graphs, *i.e.*, $\{\mathcal{Z}_i\}_{i=1}^{n}$, as a tensor $\mathcal{X} \in \mathbb{R}^{m \times m \times t \times n}$. Through joint tensor factorization, unlabeled graphs in $\mathcal{D}_u$ could facilitate the representation learning of labeled graphs in $\mathcal{D}_l$ by affecting latent factors.

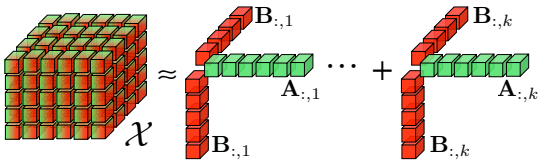


Fig. 2. CP factorization. The fourth-order partially symmetric tensor $\mathcal{X}$ is approximated by k rank-one tensors. The f-th factor tensor is the tensor product of four vectors, *i.e.*, $\mathbf{B}_{:,f} \circ \mathbf{B}_{:,f} \circ \mathbf{T}_{:,f} \circ \mathbf{A}_{:,f}$. The temporal dimension is omitted in the plot.

Note that $\mathcal{X}$ is a fourth-order partially symmetric tensor (symmetric on the first two modes) and it naturally models the temporal dimension discussed as the problem (P2). We assume that $\mathcal{X}$ can be decomposed into k factors in the following manner

$$\mathcal{X} = \mathcal{C} \times_1 \mathbf{B} \times_2 \mathbf{B} \times_3 \mathbf{T} \times_4 \mathbf{A} \tag{1}$$

where $\mathbf{B} \in \mathbb{R}^{m \times k}$ is the factor matrix for vertices, $\mathbf{T} \in \mathbb{R}^{t \times k}$ is the factor matrix for time points, $\mathbf{A} \in \mathbb{R}^{n \times k}$ is the factor matrix for graphs, and $\mathcal{C} \in \mathbb{R}^{k \times \cdots \times k}$ is a fourth-order identity tensor, *i.e.*, $\mathcal{C}(i_1, \cdots, i_4) = \delta(i_1 = \cdots = i_4)$. Basically, Eq. (1) is a CANDECOMP/PARAFAC (CP) factorization [16] as shown in Figure 2. It is desirable to discover distinct latent factors to obtain more concise and interpretable results, and thus we include orthogonality constraints $\mathbf{A}^{\mathrm{T}}\mathbf{A} = \mathbf{I}$.[1]

One of the targets is task recognition based on the brain network data. We assume that there is a matrix of regression coefficients $\mathbf{W} \in \mathbb{R}^{k \times c}$ which assigns graphs with labels based on the graph factor matrix $\mathbf{A}$, *i.e.*, $\mathbf{Y} = \mathbf{DAW}$ where $\mathbf{D} = [\mathbf{I}^{l \times l},\ \mathbf{0}^{l \times (n-l)}] \in \mathbb{R}^{l \times n}$.

An intuitive idea is to first learn latent representations of brain networks and then train a classifier on them in a serial two-step manner, which however would make these two procedures independent with each other and fail to introduce the supervision information to the representation learning process. Moreover, the advantage is established in [5] of directly searching for classification-relevant structure in the original data, rather than solving the supervised and unsupervised problems independently. To address the problem (P3) discussed in Section 1, we propose to incorporate the classifier learning process (*i.e.*, $\mathbf{W}$) into the framework of learning latent feature representations of graphs (*i.e.*, $\mathbf{A}$). In this manner, the weight matrix $\mathbf{W}$ and the feature matrix $\mathbf{A}$ can interact with each other in the same learning framework. Note that it is similar to coupled matrix and tensor factorization [2], however $\mathcal{X}$ and $\mathbf{Y}$ are coupled only in part of the graph mode, as shown in Figure 3.

[1] We considered adding non-negativity constraints to enhance interpretability, but our preliminary results showed that it would degrade performance.

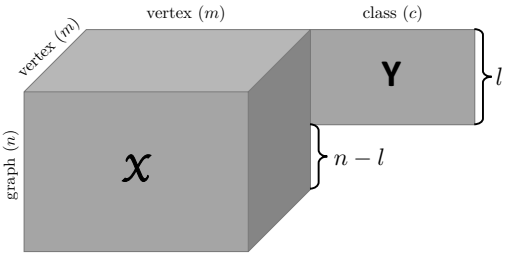


Fig. 3. Partially coupled matrix $\mathbf{Y}$ and tensor $\mathcal{X}$. The temporal dimension is omitted in the plot.

In summary, the proposed brain network analysis framework can be mathematically formulated as solving the following optimization problem

$$\min_{\mathbf{B},\mathbf{T},\mathbf{A},\mathbf{W}} \underbrace{\|\mathcal{X}-\mathcal{C}\times_1\mathbf{B}\times_2\mathbf{B}\times_3\mathbf{T}\times_4\mathbf{A}\|_F^2}_{\text{factorization error}}+\alpha\underbrace{\|\mathbf{DAW}-\mathbf{Y}\|_F^2}_{\text{classification loss}}+\lambda\underbrace{\|\mathbf{W}^{\mathrm{T}}\|_{2,1}}_{\text{regularization}}$$
$$\text{s.t. } \underbrace{\mathbf{A}^{\mathrm{T}}\mathbf{A}=\mathbf{I}}_{\text{orthogonality}} \tag{2}$$

where $\|\mathbf{W}^{\mathrm{T}}\|_{2,1}$ is the sparsity-promoting regularization term that controls the complexity of $\mathbf{W}$ and has the effects of feature selection thereby addressing the problem (P4), and α, λ are positive parameters which control contributions of classification loss and regularization, respectively.

3.3 Optimization Framework

The model parameters that have to be estimated include $\mathbf{B}\in\mathbb{R}^{m\times k}$, $\mathbf{T}\in\mathbb{R}^{t\times k}$, $\mathbf{A}\in\mathbb{R}^{n\times k}$ and $\mathbf{W}\in\mathbb{R}^{k\times c}$. The optimization problem in Eq. (2) is non-convex with respect to $\mathbf{B}$, $\mathbf{T}$, $\mathbf{A}$ and $\mathbf{W}$ together. There is no closed-form solution for the problem. We now introduce an alternating scheme to solve the optimization problem. The key idea is to optimize the objective with respect to one variable, while fixing others, and decouple constraints using an Alternating Direction Method of Multipliers (ADMM) scheme [4]. The algorithm will keep updating the variables until convergence. First, we define the following notations

$$\mathbf{E}=\mathbf{P}\odot\mathbf{T}\odot\mathbf{A}\in\mathbb{R}^{(m*t*n)\times k},\ \mathbf{F}=\mathbf{B}\odot\mathbf{T}\odot\mathbf{A}\in\mathbb{R}^{(m*t*n)\times k}$$
$$\mathbf{G}=\mathbf{B}\odot\mathbf{P}\odot\mathbf{A}\in\mathbb{R}^{(m*m*n)\times k},\ \mathbf{H}=\mathbf{B}\odot\mathbf{P}\odot\mathbf{T}\in\mathbb{R}^{(m*m*t)\times k}$$

where $\mathbf{P}\in\mathbb{R}^{m\times k}$ is the auxiliary variable.

Update the vertex factor matrix B while fixing T, A and W. Note that $\mathcal{X}$ is a partially symmetric tensor and the objective function in Eq. (2) involves a fourth-order term w.r.t. $\mathbf{B}$ which is difficult to optimize directly. To obviate this

problem, we use a variable substitution technique and minimize the following objective function

$$\begin{aligned} \min_{\mathbf{B},\mathbf{P}} \ & \|\mathbf{X}_{(1)} - \mathbf{B}\mathbf{E}^{\mathrm{T}}\|_F^2 \\ \text{s.t. } & \mathbf{P} = \mathbf{B} \end{aligned} \tag{3}$$

The augmented Lagrangian function for problem in Eq. (3) is

$$\mathcal{L}(\mathbf{B},\mathbf{P}) = \|\mathbf{X}_{(1)} - \mathbf{B}\mathbf{E}^{\mathrm{T}}\|_F^2 + \frac{\upsilon}{2}\|\mathbf{B} - \mathbf{P} - \frac{1}{\upsilon}\Upsilon\|_F^2 \tag{4}$$

where $\Upsilon \in \mathbb{R}^{m\times k}$ are Lagrange multipliers, υ is the penalty parameter which can be adjusted efficiently according to [19].

By setting the derivative of Eq. (4) w.r.t. $\mathbf{B}$ to zero, we obtain the closed-form solution

$$\mathbf{B} = (2\mathbf{X}_{(1)}\mathbf{E} + \upsilon\mathbf{P} + \Upsilon)(2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \upsilon\mathbf{I})^{-1} \tag{5}$$

To efficiently compute $\mathbf{E}^{\mathrm{T}}\mathbf{E}$, we consider the following property of the Khatri-Rao product of two matrices [16]

$$\mathbf{E}^{\mathrm{T}}\mathbf{E} = (\mathbf{P}\odot\mathbf{T}\odot\mathbf{A})^{\mathrm{T}}(\mathbf{P}\odot\mathbf{T}\odot\mathbf{A}) = \mathbf{P}^{\mathrm{T}}\mathbf{P} * \mathbf{T}^{\mathrm{T}}\mathbf{T} * \mathbf{A}^{\mathrm{T}}\mathbf{A} \tag{6}$$

Similarly, the auxiliary matrix $\mathbf{P}$ can be optimized successively

$$\mathbf{P} = (2\mathbf{X}_{(2)}\mathbf{F} + \upsilon\mathbf{B} - \Upsilon)(2\mathbf{F}^{\mathrm{T}}\mathbf{F} + \upsilon\mathbf{I})^{-1} \tag{7}$$

The Lagrange multipliers Υ can be updated using gradient ascent

$$\Upsilon \leftarrow \Upsilon + \upsilon(\mathbf{P} - \mathbf{B}) \tag{8}$$

Update the temporal factor matrix T while fixing B, A and W. Since there is no constraint on $\mathbf{T}$, we directly obtain the closed-form solution

$$\mathbf{T} = (\mathbf{X}_{(3)}\mathbf{G})(\mathbf{G}^{\mathrm{T}}\mathbf{G})^{-1} \tag{9}$$

Update the graph factor matrix A while fixing B, T and W. By variable substitution, we need to minimize the following objective function

$$\begin{aligned} \min_{\mathbf{A},\mathbf{Q}} \ & \|\mathbf{X}_{(4)} - \mathbf{A}\mathbf{H}^{\mathrm{T}}\|_F^2 + \alpha\|\mathbf{D}\mathbf{A}\mathbf{W} - \mathbf{Y}\|_F^2 \\ \text{s.t. } & \mathbf{Q}^{\mathrm{T}}\mathbf{A} = \mathbf{I},\ \mathbf{Q} = \mathbf{A} \end{aligned} \tag{10}$$

The augmented Lagrangian function for problem in Eq. (10) is

$$\begin{aligned} \mathcal{L}(\mathbf{A},\mathbf{Q}) = {} & \|\mathbf{X}_{(4)} - \mathbf{A}\mathbf{H}^{\mathrm{T}}\|_F^2 + \alpha\|\mathbf{D}\mathbf{A}\mathbf{W} - \mathbf{Y}\|_F^2 \\ & + \frac{\phi}{2}\|\mathbf{A} - \mathbf{Q} - \frac{1}{\phi}\Phi\|_F^2 + \frac{\psi}{2}\|\mathbf{I} - \mathbf{Q}^{\mathrm{T}}\mathbf{A} - \frac{1}{\psi}\Psi\|_F^2 \end{aligned} \tag{11}$$

where $\Phi \in \mathbb{R}^{n\times k}$ and $\Psi \in \mathbb{R}^{k\times k}$ are Lagrange multipliers, ϕ and ψ are penalty parameters. By setting the derivative of Eq. (11) w.r.t. $\mathbf{A}$ to zero, we obtain the Sylvester equation

$$\begin{aligned} &X\mathbf{A} + \mathbf{A}Y = Z \\ &X = \psi\mathbf{Q}\mathbf{Q}^{\mathrm{T}} \\ &Y = 2\mathbf{H}^{\mathrm{T}}\mathbf{H} + 2\alpha\mathbf{W}\mathbf{W}^{\mathrm{T}} + \phi\mathbf{I} \\ &Z = 2\mathbf{X}_{(4)}\mathbf{H} + 2\alpha\mathbf{D}^{\mathrm{T}}\mathbf{Y}\mathbf{W}^{\mathrm{T}} + (\phi+\psi)\mathbf{Q} + \Phi - \mathbf{Q}\Psi \end{aligned} \tag{12}$$

which can be solved by several numerical approaches, *e.g.*, the *lyap* function in MATLAB.

The closed-form update for $\mathbf{Q}$ is

$$\mathbf{Q} = (\psi\mathbf{A}\mathbf{A}^{\mathrm{T}} + \phi\mathbf{I})^{-1}((\phi+\psi)\mathbf{A} - \Phi - \mathbf{A}\Psi^{\mathrm{T}}) \tag{13}$$

The Lagrange multipliers Φ and Ψ can be updated by

$$\Phi \leftarrow \Phi + \phi(\mathbf{Q} - \mathbf{A}),\ \Psi \leftarrow \Psi + \psi(\mathbf{Q}^{\mathrm{T}}\mathbf{A} - \mathbf{I}) \tag{14}$$

Update the weight matrix W while fixing B, T and A. According to the analysis of the $\ell_{2,1}$ norm in [21], we need to minimize the following objective function

$$\mathcal{L}(\mathbf{W}) = \|\mathbf{D}\mathbf{A}\mathbf{W} - \mathbf{Y}\|_F^2 + \gamma\|\mathbf{W}^{\mathrm{T}}\|_{2,1} = \|\mathbf{D}\mathbf{A}\mathbf{W} - \mathbf{Y}\|_F^2 + \gamma\mathrm{tr}(\mathbf{W}\Omega\mathbf{W}^{\mathrm{T}}) \tag{15}$$

where $\Omega \in \mathbb{R}^{c\times c}$ is an auxiliary diagonal matrix of the $\ell_{2,1}$ norm. The diagonal elements of Ω are computed as $\Omega(i,i) = \frac{1}{2\sqrt{||\mathbf{W}(:,i)||_2^2+\epsilon}}$ where ϵ is a smoothing term which is usually set to a small constant.

By setting the derivative of Eq. (15) w.r.t. $\mathbf{W}$ to zero, we obtain the Sylvester equation

$$\begin{aligned} &X\mathbf{W} + \mathbf{W}Y = Z \\ &X = 2\mathbf{A}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{D}\mathbf{A} \\ &Y = \gamma\Omega \\ &Z = 2\mathbf{A}^{\mathrm{T}}\mathbf{D}^{\mathrm{T}}\mathbf{Y} \end{aligned} \tag{16}$$

Based on the above analysis, we develop the optimization framework for brain network analysis based on tensor factorization, as described in Algorithm 1. The code has been made available at the author's homepage[2].

3.4 Time Complexity

Each ADMM iteration consists of simple matrix operations. Therefore, rough estimates of its computational complexity can be easily derived [18].

[2] `https://www.cs.uic.edu/~bcao1/code/semibat.zip`

Algorithm 1 semiBAT

Input: $\mathcal{X}, \mathbf{Y}, \alpha, \lambda$
Output: $\mathbf{B}, \mathbf{T}, \mathbf{A}, \mathbf{W}$
1: Set $\upsilon_{max} = \phi_{max} = \psi_{max} = 10^6, \rho = 1.15$
2: Initialize $\mathbf{B}, \mathbf{T}, \mathbf{A}, \mathbf{W} \sim \mathcal{U}(0,1), \Upsilon = \Phi = \Psi = \mathbf{0}, \upsilon = \phi = \psi = 10^{-6}$
3: **repeat**
4: Update $\mathbf{B}$ and $\mathbf{P}$ by Eq. (5) and Eq. (7)
5: Update $\mathbf{T}$ by Eq. (9)
6: Update $\mathbf{A}$ and $\mathbf{Q}$ by Eq. (12) and Eq. (13)
7: Update $\mathbf{W}$ by Eq. (16)
8: Update Υ, Φ and Ψ by Eq. (8) and Eq. (14)
9: $\upsilon \leftarrow \min(\rho\upsilon, \upsilon_{max})$, $\phi \leftarrow \min(\rho\phi, \phi_{max})$, $\psi \leftarrow \min(\rho\psi, \psi_{max})$
10: **until** convergence

- The estimate for the update of $\mathbf{B}$ according to Eq. (5) is: 1) $O(m^2ntk)$ for the computation of the term $2\mathbf{X}_{(1)}\mathbf{E} + \upsilon\mathbf{P} + \Upsilon$, 2) $O((m+n+t)k^2)$ for the computation of the term $2\mathbf{E}^{\mathrm{T}}\mathbf{E} + \upsilon\mathbf{I}$ due to Eq. (6), $O(k^3)$ for its Cholesky decomposition, and 3) $O(mk^2)$ for the computation of the system solution that gives the updated value of $\mathbf{B}$. An analogous estimate can be derived for the update of $\mathbf{P}$ and $\mathbf{T}$ which cost $O(k^3 + (m+n+t)k^2 + m^2ntk)$.
- Considering $l < n$ and c is usually a small constant, the estimate for the update of $\mathbf{A}$ according to Eq. (12) is: 1) $O(n^2k)$ for the computation of the term X, 2) $O((m+n+t)k^2)$ for the computation of the term Y, 3) $O(nk^2 + m^2ntk + n^2)$ for the computation of the term Z, and 4) $O(nk^2 + n^2k)$ for the computation of the Sylvester equation [14].
- The estimate for the update of $\mathbf{Q}$ according to Eq. (13) is $O(nk^2 + n^2k + n^3)$.
- The estimate for the update of $\mathbf{W}$ according to Eq. (16) is $O(nk^2 + n^2k)$.

Overall, the updates of all model parameters require $O(k^3 + (m+n+t)k^2 + (m^2nt + n^2)k + n^3)$ arithmetic operations in total.

4 Experiments

4.1 Data Collections

Data were collected from 22 healthy participants at the University of Illinois at Chicago (UIC) and from 11 healthy participants at the University of Michigan (UMich), respectively. Each participant underwent an emotion regulation task, while UIC participants further underwent an eight-minute resting-state recording session which served as unlabeled data. During the ERT session, participants were instructed to look at pictures displayed on the screen. Emotionally neutral pictures (*e.g.*, landscape, everyday objects) and negative pictures (*e.g.*, car crash, natural disasters) would appear on the screen for seven seconds in random orders. One second after the picture on display, a corresponding auditory guide would instruct the participant to *look*: viewing the neutral pictures; to *maintain*:

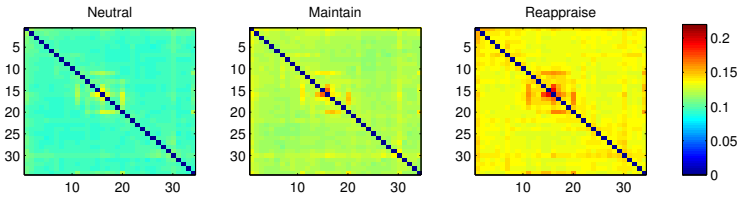


Fig. 4. Average brain networks during NEUTRAL, MAINTAIN and REAPPRAISE.

viewing the negative pictures as they normally would; or to *reappraise*: viewing the negative pictures while attempting to reduce their emotion response by re-interpreting the meaning of pictures. All EEG data were recorded using the Biosemi system equipped with an elastic cap with 34 scalp channels. A detailed description about data acquisition and preprocessing is available in [28].

Overall, the dataset contains $n = 121$ EEG brain network samples that are based upon $m = 34$ vertices and $t = 130$ time points. The target is to train a classifier on the UIC source (66 training samples and 22 unlabeled samples) to predict which task (NEUTRAL, MAINTAIN, or REAPPRAISE) a subject in the UMich source (33 test samples) is performing. The average brain networks are shown in Figure 4 where the x and y axes represent the vertex id, and the color of the cell represents the strength of the connection between vertex x and y. Although the group difference appears to be significant, it is non-trivial to identify the tasks for each individual. It will be validated in the experiments that simply using edge values as features to train a classifier could not lead to a good classification performance.

4.2 Compared Methods

The compared methods are summarized as follows:

- SEMIBAT: the proposed semi-supervised brain network analysis approach based on constrained tensor factorization.
- BAT-RIDGE: replacing the $\ell_{2,1}$ norm in SEMIBAT with a regular ridge term.
- BAT-SUPV: a fully supervised variant of SEMIBAT without leveraging the unlabeled data.
- BAT-UNSUPV: an unsupervised variant of SEMIBAT that first learns latent representations of brain networks and then trains a classifier on them in a serial two-step manner.
- BAT-3D: applying SEMIBAT on time-averaged brain networks.
- ALS: plain vanilla tensor factorization using alternating least squares without any constraint [8].
- SUBGRAPH: a discriminative subgraph selection method for uncertain graph classification [17,7].
- CC: extracting local clustering coefficients as features, one of the most popular graph-theoretical measures that quantify the cliquishness of the vertices [23].

Table 2. Classification performance. *N*, *M* and *R* stand for tasks: NEUTRAL, MAINTAIN and REAPPRAISE, respectively. The best performance on each metric is in bold.

Methods	Evaluation metrics							
	Accuracy	Precision			Recall			F1
		N	*M*	*R*	*N*	*M*	*R*	
semiBAT	**0.758**	0.833	**0.889**	0.667	**0.833**	0.667	**0.833**	**0.765**
BAT-ridge	0.697	**0.909**	0.700	0.600	**0.833**	0.583	0.750	0.706
BAT-supv	0.697	0.714	0.750	**0.700**	**0.833**	**0.750**	0.583	0.706
BAT-unsupv	0.576	0.818	0.600	0.467	0.750	0.500	0.583	0.588
BAT-3d	0.545	0.857	0.538	0.500	0.500	0.583	0.667	0.559
ALS	0.576	0.750	0.636	0.462	0.750	0.583	0.500	0.588
Subgraph	0.515	0.800	0.500	0.444	0.667	0.333	0.667	0.529
CC	0.364	0.286	0.667	0.391	0.167	0.333	0.750	0.382
Edge	0.455	0.462	0.700	0.385	0.500	0.583	0.417	0.471

– EDGE: using edge values as features by flatting adjacency matrices of brain networks into vectors.

For a fair comparison, we used a regularized regression in semiBAT as the base classifier for all the compared methods. The parameters α and λ were tuned in the range of $2^{-10}, ..., 2^{10}$, the rank k was tuned in the range of $1, ..., 20$. The accuracy with the best parameter configuration was reported, as well as the corresponding precision, recall and F1 score.

4.3 Classification Performance

Experimental results in Table 2 show the classification performance of compared methods on distinguishing the three tasks. EDGE serves as the basis for comparison that treats a brain network as a collection of edges, thereby blinding the connectivity structures of brain networks, which surprisingly outperforms CC. Although clustering coefficients have been widely used to identify Alzheimer's disease [27,13], they appear to be less useful for distinguishing the emotion regulation tasks. SUBGRAPH achieves a better performance by extracting connectivity patterns within brain networks.

Factorization models demonstrate themselves with significantly better accuracy. According to the low-rank assumption, a low-dimensional latent factor of each graph is obtained by first stacking all the brain network data and then factorizing the constructed tensor. ALS is a direct application of the alternating least squares technique to the standard tensor factorization problem without incorporating any constraint or supervision. A significant improvement of 31.60% by semiBAT over ALS can be observed, mainly due to the fact that the unsupervised ALS approach fails to interact with the classifier training procedure which shows comparable performance with BAT-unsupv. It indicates the importance

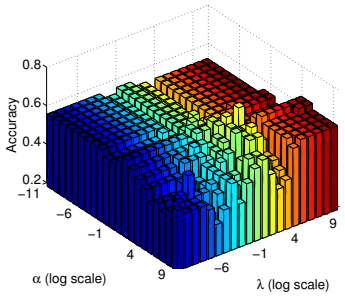


Fig. 5. Sensitivity w.r.t. α and λ.

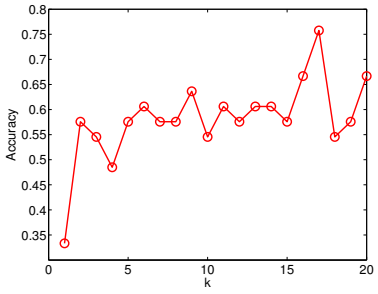


Fig. 6. Sensitivity w.r.t. k.

of addressing the problem (P3) discussed in Section 1. Moreover, semiBAT outperforms BAT-ridge thereby demonstrating that it is critical to apply feature selection in the tensor factorization framework (*i.e.*, the problem (P4)). The advantages of semiBAT over BAT-supv and BAT-3d are attributed to leveraging unlabeled resting-state brain network data (*i.e.*, the problem (P1)) and modeling the temporal dimension (*i.e.*, the problem (P2)), respectively.

4.4 Parameter Sensitivity

In all experiments, the regularization parameter λ was tuned for all the baselines, the rank k was tuned for all the factorization models, and α was tuned for semiBAT and its variants. We first investigate the influence of α and λ in semiBAT and present the results in Figure 5. It illustrates that neither a small nor a large α or λ would be preferred, and in general, a good choice of α and λ can be found in the range of $2^5, ..., 2^7$ and $2^0, ..., 2^2$, respectively. Moreover, experimental results of factorization models with different k are shown in Figure 6. In general, a small k would rarely be a wise choice, and the best performance can usually be achieved around $k = 17$.

4.5 Factor Analysis

We first investigate the factor matrices derived from semiBAT in a row-wise manner. Note that initially with the best parameter configuration as reported in the last section where $k = 17$, we obtain a 17-dimensional feature vector for each brain network (*i.e.*, $\mathbf{A}(i,:)$) and each time point (*i.e.*, $\mathbf{T}(i,:)$). For visualization we use t-SNE [25] to reduce them into a 2-dimensional space. In Figure 7, we show the distribution of brain networks, where there are 99 points representing 33 samples from each of the three tasks (22 resting-state samples are omitted). A relatively clear separation between Neutral and Reappraise can be observed, while Maintain usually mix with the other two conditions which make the classification problem challenging. Figure 8 illustrates the distribution of time points, where there are 130 points and each of them represents an exact time point indicated by the color. Basically, adjacent time points are colored similarly.

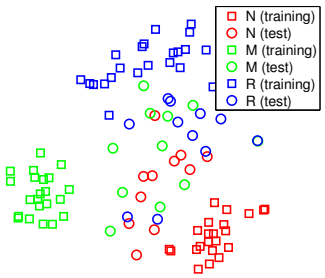


Fig. 7. Embedding of brain networks.

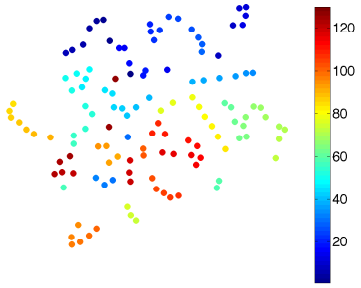
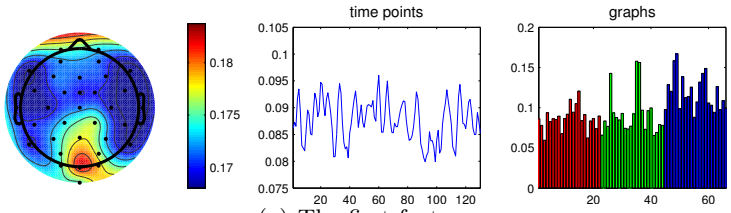
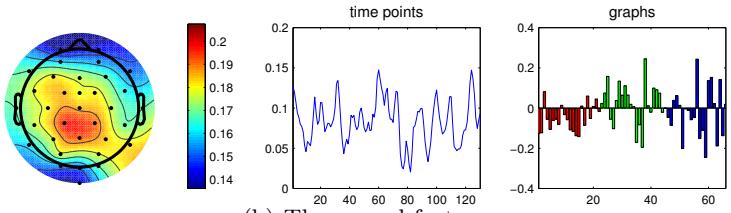


Fig. 8. Embedding of time points.



(a) The first factor.



(b) The second factor.

Fig. 9. The two largest factors in terms of magnitude derived from SEMIBAT model for task recognition.

From this figure, we can see that continuous time points form distinct clusters and brain activities change over time, so it is important to capture the temporal dimension explicitly.

Next, we visualize and interpret the factor matrices in a column-wise manner. A k-factor SEMIBAT model extracts the factors $\mathbf{B}(:, i)$, $\mathbf{T}(:, i)$ and $\mathbf{A}(:, i)$, for $i = 1, ..., k$, where these factors indicate the signatures of sources in vertex, time and graph domain, respectively. We show the two largest factors in terms of magnitude in Figure 9. In the left panel, points indicate the spatial layout of electrodes (*i.e.*, vertices) on the scalp, and factor values of electrodes are demonstrated on a colormap using EEGLAB [10]. The middle panel shows the temporal changes of the factor. The right panel shows the strength of 66 brain networks in the training set performing different tasks where red, green and blue stand for NEUTRAL, MAINTAIN and REAPPRAISE, respectively. A domain

expert can identify the brain activity pattern in the left panel, the corresponding coefficients of time points in the middle panel, and the graph difference on such pattern in the right panel. We can see that different latent factors capture activity of different brain regions. The first factor appears to highlight a quantitative anterior-posterior gradient (maximum values of the first factor appear in the occipital lobe) that is shared across all three conditions, thus may be related to visual processing, while the second factor, which primarily differentiates neutral from maintain and reappraisal, predominantly involves electrodes around the frontal-parietal junction and thus may be related to the late positive potential [11,12].

5 Related Work

Tensor factorization has become an effective technique in many healthcare applications. For example, Acar et al. identify spatial, spectral and temporal signatures of an epileptic seizure as well as an artifact through the application of tensor models [1]. Davidson et al. propose a constrained alternating least squares framework for network discovery of fMRI data [9]. Papalexakis et al. present a scalable solution for the coupled matrix-tensor factorization problem, and find latent variables that jointly explain both the brain activity and the behavioral responses [22]. Wang et al. introduce knowledge guided tensor factorization for computational phenotyping [26]. Ma et al. propose a spatio-temporal tensor kernel approach for whole-brain fMRI image analysis [20]. However, these frameworks are not directly applicable to partially symmetric tensor factorization or further task recognition.

For graph classification on brain networks, literatures have been focused on first deriving vector presentations from the brain network data which are then fed into conventional pattern classifiers. In general, two types of features are usually extracted: (1) graph-theoretical measures and (2) subgraph patterns. Wee et al. extract weighted local clustering coefficients of each brain region in relation to other regions in brain networks to quantify the prevalence of clustered connectivity around brain regions for diagnosis on Alzheimer's disease [27]. In addition to the local network property, Jie et al. use a topology-based graph kernel to measure the topological similarity between paired fMRI brain networks [13]. Kong et al. propose a discriminative subgraph feature selection method based on dynamic programming to compute the probability distribution of discrimination scores for each subgraph pattern within a set of weighted graphs [17]. In contrast to focusing on the graph view alone, Cao et al. introduce a subgraph mining algorithm using side information guidance to find an optimal set of subgraph features for graph classification [6]. However, the expressiveness of these features is limited to the predefined formulations. It is critical to explore a larger space of potentially informative features to represent brain networks through data-driven approaches.

6 Conclusion

This paper presents SEMIBAT, a novel semi-supervised brain network analysis approach based on constrained tensor factorization. It leverages unlabeled resting-state brain networks for task recognition, explores the temporal dimension to capture the progress, incorporates classifier learning procedure to introduce supervision from labeled data, and selects discriminative latent factors for different tasks. ADMM is used to solve the optimization problem. In the experiments on EEG datasets, we demonstrate the superior performance of SEMIBAT on graph classification tasks over the state-of-art methods.

Acknowledgements This work is supported in part by NSF through grants III-1526499.

References

1. Evrim Acar, Canan Aykut-Bingol, Haluk Bingol, Rasmus Bro, and Bülent Yener. Multiway analysis of epilepsy tensors. *Bioinformatics*, 23(13):i10–i18, 2007.
2. Evrim Acar, Tamara G Kolda, and Daniel M Dunlavy. All-at-once optimization for coupled matrix and tensor factorizations. *arXiv:1105.3422*, 2011.
3. Olusola Ajilore, Liang Zhan, Johnson GadElkarim, Aifeng Zhang, Jamie D Feusner, Shaolin Yang, Paul M Thompson, Anand Kumar, and Alex Leow. Constructing the resting state structural connectome. *Frontiers in neuroinformatics*, 7, 2013.
4. Stephen Boyd, Neal Parikh, Eric Chu, Borja Peleato, and Jonathan Eckstein. Distributed optimization and statistical learning via the alternating direction method of multipliers. *Foundations and Trends® in Machine Learning*, 3(1):1–122, 2011.
5. Danilo Bzdok, Michael Eickenberg, Olivier Grisel, Bertrand Thirion, and Gaël Varoquaux. Semi-supervised factored logistic regression for high-dimensional neuroimaging data. In *NIPS*, pages 3330–3338, 2015.
6. Bokai Cao, Xiangnan Kong, Jingyuan Zhang, Philip S. Yu, and Ann B. Ragin. Mining brain networks using multiple side views for neurological disorder identification. In *ICDM*, pages 709–714. IEEE, 2015.
7. Bokai Cao, Liang Zhan, Xiangnan Kong, Philip S. Yu, Nathalie Vizueta, Lori L. Altshuler, and Alex D. Leow. Identification of discriminative subgraph patterns in fMRI brain networks in bipolar affective disorder. In *Brain Informatics and Health*, pages 105–114. Springer, 2015.
8. Pierre Comon, Xavier Luciani, and André LF De Almeida. Tensor decompositions, alternating least squares and other tales. *Journal of Chemometrics*, 23(7-8):393–405, 2009.
9. Ian Davidson, Sean Gilpin, Owen Carmichael, and Peter Walker. Network discovery via constrained tensor analysis of fMRI data. In *KDD*, pages 194–202. ACM, 2013.
10. Arnaud Delorme and Scott Makeig. EEGLAB: an open source toolbox for analysis of single-trial EEG dynamics including independent component analysis. *Journal of neuroscience methods*, 134(1):9–21, 2004.
11. Tracy A Dennis and Greg Hajcak. The late positive potential: a neurophysiological marker for emotion regulation in children. *Journal of Child Psychology and Psychiatry*, 50(11):1373–1383, 2009.
12. Greg Hajcak, Annmarie MacNamara, and Doreen M Olvet. Event-related potentials, emotion, and emotion regulation: an integrative review. *Developmental neuropsychology*, 35(2):129–155, 2010.

13. Biao Jie, Daoqiang Zhang, Wei Gao, Qian Wang, C Wee, and Dinggang Shen. Integration of network topological and connectivity properties for neuroimaging classification. *Biomedical Engineering*, 61(2):576, 2014.
14. Isak Jonsson and Bo Kågström. Recursive blocked algorithms for solving triangular systemspart i: One-sided and coupled sylvester-type matrix equations. *ACM Transactions on Mathematical Software*, 28(4):392–415, 2002.
15. Junghoe Kim, Vince D Calhoun, Eunsoo Shim, and Jong-Hwan Lee. Deep neural network with weight sparsity control and pre-training extracts hierarchical features and enhances classification performance: Evidence from whole-brain resting-state functional connectivity patterns of schizophrenia. *NeuroImage*, 124:127–146, 2016.
16. Tamara G Kolda and Brett W Bader. Tensor decompositions and applications. *SIAM review*, 51(3):455–500, 2009.
17. Xiangnan Kong, Philip S Yu, Xue Wang, and Ann B Ragin. Discriminative feature selection for uncertain graph classification. In *SDM*, 2013.
18. Athanasios P Liavas and Nicholas D Sidiropoulos. Parallel algorithms for constrained tensor factorization via the alternating direction method of multipliers. *arXiv:1409.2383*, 2014.
19. Zhouchen Lin, Risheng Liu, and Zhixun Su. Linearized alternating direction method with adaptive penalty for low-rank representation. In *NIPS*, pages 612–620, 2011.
20. Guixiang Ma, Lifang He, Chun-Ta Lu, Philip S. Yu, Linlin Shen, and Ann B. Ragin. Spatio-temporal tensor analysis for whole-brain fMRI classification. In *SDM*. SIAM, 2016.
21. Feiping Nie, Heng Huang, Xiao Cai, and Chris H Ding. Efficient and robust feature selection via joint 2, 1-norms minimization. In *NIPS*, pages 1813–1821, 2010.
22. Evangelos E Papalexakis, Tom M Mitchell, Nicholas D Sidiropoulos, Christos Faloutsos, Partha Pratim Talukdar, and Brian Murphy. Turbo-smt: Accelerating coupled sparse matrix-tensor factorizations by 200x. In *SDM*. SIAM, 2014.
23. Mikail Rubinov and Olaf Sporns. Complex network measures of brain connectivity: uses and interpretations. *Neuroimage*, 52(3):1059–1069, 2010.
24. Olaf Sporns, Giulio Tononi, and Rolf Kötter. The human connectome: a structural description of the human brain. *PLoS computational biology*, 1(4):e42, 2005.
25. Laurens Van der Maaten and Geoffrey Hinton. Visualizing data using t-sne. *Journal of Machine Learning Research*, 9(2579-2605):85, 2008.
26. Yichen Wang, Robert Chen, Joydeep Ghosh, Joshua C Denny, Abel Kho, You Chen, Bradley A Malin, and Jimeng Sun. Rubik: Knowledge guided tensor factorization and completion for health data analytics. In *KDD*, pages 1265–1274. ACM, 2015.
27. Chong-Yaw Wee, Pew-Thian Yap, Daoqiang Zhang, Kevin Denny, Jeffrey N Browndyke, Guy G Potter, Kathleen A Welsh-Bohmer, Lihong Wang, and Dinggang Shen. Identification of MCI individuals using structural and functional connectivity networks. *Neuroimage*, 59(3):2045–2056, 2012.
28. Mengqi Xing, Reza Tadayonnejad, Annmarie MacNamara, Olusola Ajilore, K. Luan Phan, Heide Klumpp, and Alex Leow. EEG based functional connectivity reflects cognitive load during emotion regulation. In *ISBI*. IEEE, 2016.
29. Jieping Ye, Kewei Chen, Teresa Wu, Jing Li, Zheng Zhao, Rinkal Patel, Min Bae, Ravi Janardan, Huan Liu, Gene Alexander, and Eric Reiman. Heterogeneous data fusion for Alzheimer's disease study. In *KDD*, pages 1025–1033. ACM, 2008.
30. Jingyuan Zhang, Bokai Cao, Sihong Xie, Chun-Ta Lu, Philip S. Yu, and Ann B. Ragin. Identifying connectivity patterns for brain diseases via multi-side-view guided deep architectures. In *SDM*. SIAM, 2016.