Learning from Brain Data for Neurological Disorder Analysis

BY

GUIXIANG MA
M.E., Beijing Jiaotong University, 2013
B.S., Liaoning Normal University, 2010

THESIS

Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Computer Science
in the Graduate College of the
University of Illinois at Chicago, 2019

Chicago, Illinois

Defense Committee:
Philip S. Yu, Chair and Advisor
Bing Liu
Xinhua Zhang
Yuheng Hu, Dept. of Information and Decision Sciences
Ann B. Ragin, Northwestern University

This dissertation is dedicated to my parents

for their love, encouragement and endless support.

ACKNOWLEDGMENTS

First and foremost, I would like to express my sincere gratitude to my Ph.D. advisor, Prof. Philip S. Yu, for his precious guidance and persistent encouragement along the way. Without his continuous support, this dissertation would not have been possible.

I would like to thank Dr. Ann B. Ragin at Northwestern University for her patient mentoring and valuable suggestions during our collaborations in various projects that relate to this dissertation. Besides, I would like to thank the rest of my dissertation committee: Prof. Bing Liu, Prof. Xinhua Zhang and Prof. Yuheng Hu for taking their valuable time to serve as my dissertation committee members. I appreciate their insightful suggestions and crucial remarks that shaped this dissertation.

My sincere thanks also goes to Prof. Youli Qu at Beijing Jiaotong University, who mentored and guided me through the early stage of my research career. I also want to thank Dr. Ted Willke, Dr. Nesreen K. Ahmed and Dr. Dipanjan Sengupta at Intel Labs, who mentored me during my research internship at the Brain-inspired Computing Lab at Intel.

I would like to extend my warmest thanks to all the colleagues and friends that I met at the University of Illinois at Chicago. I am grateful to these great people for the inspiring discussions, collaborations and the countless happy moments we have experienced together.

Finally, I would like to thank my parents and my sister for their endless love and support in whatever I pursue all these years.

GM

CONTRIBUTION OF AUTHORS

Chapter 1 is an introduction that outlines my dissertation research. Chapter 2 presents a published manuscript (Ma et al., 2016) for which I was the primary author. Dr. Lifang He contributed to the tensor modeling and drafting a part of the manuscript. Dr. Chun-Ta Lu, Prof. Philip S. Yu, Linlin Shen and Dr. Ann B. Ragin contributed to discussions with respect to the work and revising the manuscript.

Chapter 3 presents a published manuscript (Ma et al., 2016), for which I was the primary author. Dr. Lifang He contributed to the optimization techniques. Dr. Bokai Cao, Dr. Jiawei Zhang, Prof. Philip S. Yu and Dr. Ann B. Ragin contributed to the discussions with respect to the work and revising the manuscript.

Chapter 4 presents a published manuscript (Ma et al., 2017) for which I was the primary author. Dr. Lifang He, Dr. Chun-Ta Lu, Dr. Weixiang Shao, Prof. Philip S. Yu, Dr. Alex D. Leow and Dr. Ann B. Ragin contributed to discussions with respect to the work and revising the manuscript.

Chapter 5 presents a published manuscript (Ma et al., 2017) for which I was the primary author. Dr. Chun-Ta Lu, Dr. Lifang He, Prof. Philip S. Yu and Dr. Ann B. Ragin contributed to discussions with respect to the work and revising the manuscript.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

CHAPTER **PAGE**

LIST OF TABLES

LIST OF FIGURES

SUMMARY

In recent years, the advancement in neuroimaging technology has given rise to various modalities of brain imaging data, which provides us with unprecedented opportunities for investigating the inner organization and activity of human brain for neurological disorder analysis. These brain data can be acquired in different forms, such as the spatio-temporal tensor data (*e.g.*, fMRI 4D tensor image), graph data (*e.g.*, fMRI brain connectivity networks) and multi-view graph data (*e.g.*, fMRI and DTI brain networks). Learning from these brain data and leveraging the information for neurological disorder analysis can potentially facilitate the clinical investigation and therapeutic intervention of many brain diseases.

In this dissertation, I will introduce our recent works on modeling and learning from brain data in multiple perspectives for neurological disorder analysis. In the first part, I focus on the spatio-temporal tensor modeling of fMRI image data for whole-brain classification. In the second part, I present an approach based on interior-node topology of graphs for the clustering of brain networks. In the third part, a multi-view clustering framework is proposed with graph embedding for the clustering of multi-view brain networks. In the fourth part, I introduce a multi-view graph embedding approach for jointly learning the multi-view representation and detecting hubs from multi-view brain networks.

CHAPTER 1

INTRODUCTION

1.1 Dissertation Outline

With the development of neuroimaging technology, various modalities of brain imaging data have become available. These brain data encodes tremendous information about the inner organization and activities of human brain. For example, functional magnetic resonance imaging (fMRI) can be used to study the functional activation patterns of human brain based on the cerebral blood flow and the BOLD response, while diffusion tensor imaging (DTI) can be used for examining the tractograph of the white matter fiber pathways and thus for exploring the structural connectivity in the brain. These structural and functional information could reflect the neurological health status of individuals, thus could be used for neurological disorder diagnosis. However, the brain data can be acquired in different forms, such as the spatio-temporal tensor data (*e.g.*, fMRI 4D tensor image), graph data (*e.g.*, fMRI brain connectivity networks) and multi-view graph data (*e.g.*, fMRI and DTI brain networks). How to learn discriminative and meaningful representations from the different forms and modalities of brain data for neurological disorder analysis is a critical problem.

This dissertation focuses on modeling and learning from brain data in multiple perspectives for neurological disorder analysis. Specifically, it involves analysis on spatio-temporal fMRI

tensor images and brain connectivity networks derived from multiple modalities of brain imaging data. Four different learning tasks related to the above analysis are covered in this dissertation:

- To learn discriminative representations from spatio-temporal fMRI data, we propose a spatio-temporal tensor kernel (STTK) approach with time series extraction and tensor factorization for whole-brain fMRI image classification.
- We provide a framework for clustering brain networks based on interior-node topology, where the topological structure of each brain network is extracted through a sparsity-inducing interior-node clustering procedure.
- To integrate multiple views of brain networks and take advantages of their consensus and complimentary information, we incorporate multi-view graph embedding into the clustering problem and propose an approach for multi-view clustering of brain networks.
- We propose an auto-weighted framework of multi-view graph embedding with hub Detection (MVGE-HD) for multi-view brain network analysis.

1.2 Spatio-Temporal Tensor Analysis

(Part of this chapter was previously published in (Ma et al., 2016).)

Owing to prominence as a research and diagnostic tool in human brain mapping, whole-brain fMRI image analysis has been the focus of intense investigation. Conventionally, input fMRI brain images are converted into vectors or matrices and adapted in kernel based classifiers. fMRI data, however, are inherently coupled with sophisticated spatio-temporal tensor structure (*i.e.*, 3D space $\times$ time). Valuable structural information will be lost if the tensors are

converted into vectors. Furthermore, time series fMRI data are noisy, involving time shift and low temporal resolution. To address these analytic challenges, more compact and discriminative representations for kernel modeling are needed.

In Chapter 2, we propose a novel spatio-temporal tensor kernel (STTK) approach for whole-brain fMRI image analysis. Specifically, we design a volumetric time series extraction approach to model the temporal data, and propose a spatio-temporal tensor based factorization for feature extraction. We further leverage the tensor structure to encode prior knowledge in the kernel.

1.3 Multi-graph Clustering

(Part of this chapter was previously published in (Ma et al., 2016))

From brain images such as the functional magnetic resonance imaging (fMRI) data of multiple subjects, we can construct a brain connectivity network for each of them, where each node represents a brain region, and each link represents the functional/structural connectivity between two brain regions (Kong and Yu, 2014). These multiple brain networks indicate the activity patterns of each brain regions and also reflect the collaborations among different regions of the human brain, serving as valuable supportive information for clinical diagnosis of neurological disorders (Ragin et al., 2012a).

In Chapter 3, we investigate the unsupervised scenarios by exploring the multi-graph clustering for brain network clustering analysis. A multi-graph clustering approach (MGCT) is proposed based on the interior-node topology of graphs. Specifically, we extract the interior-node topological structure of each graph through a sparsity-inducing interior-node clustering. We merge the interior-node clustering stage and the multi-graph clustering stage into a unified

iterative framework, where the multi-graph clustering will influence the interior-node clustering and the updated interior-node clustering results will be further exerted on multi-graph clustering. This framework enables both the subject clustering analysis and the group-contrasting interior-node structural analysis.

1.4 Multi-view Clustering with Graph Embedding

(Part of this chapter was previously published in (Ma et al., 2017).) Multi-view clustering has become a widely studied problem in the area of unsupervised learning. It aims to integrate multiple views by taking advantages of the consensus and complimentary information from multiple views. Most of the existing works in multi-view clustering utilize the vector-based representation for features in each view. However, in many real-world applications like brain network analysis, instances are represented by graphs, where those vector-based models cannot fully capture the structure of the graphs from each view.

To solve this problem, in Chapter 4 we propose a Multi-view Clustering framework on graph instances with Graph Embedding (MCGE) for multi-view brain network analysis. Specifically, we model the multi-view graph data as tensors and apply tensor factorization to learn the multi-view graph embeddings, thereby capturing the local structure of graphs. We build an iterative framework by incorporating multi-view graph embedding into the multi-view clustering task on graph instances, jointly performing multi-view clustering and multi-view graph embedding simultaneously. The multi-view clustering results are used for refining the multi-view graph embedding, and the updated multi-view graph embedding results further improve the multi-view clustering. We apply this framework to study the connectome of fMRI and DTI brain

networks, which enable us to cluster the subjects with similar neurological status into the same group according to their brain connectivity in both views.

1.5 Multi-view Graph Embedding with Hub Detection

(Part of this chapter was previously published in (Ma et al., 2017).)

Multi-view graph embedding, as a hot topic in multi-view graph learning, has drawn extensive attentions in the past decade. Most of the existing works in multi-view graph embedding aim to combine the information from all the views and obtain a lower dimensional but better feature representation of the nodes for the spectral clustering problem (Kumar et al., 2011; Papalexakis et al., 2013; Cai et al., 2011). Although these works can be used to obtain the graph embeddings from multiple views, none of them has considered the hubs when learning the multi-view graph embedding, making them less capable for the scenarios where hubs are also important for the clustering of nodes in graphs. Specifically, in neuroscience studies, the hubs of brain networks have been proven to be more biologically costly due to higher blood flow or connection distances, thus they tend to be more vulnerable to brain injuries (Crossley et al., 2014). As a result, the hubs will differ in the brain networks of normal people and those of the subjects with neurological disorder, which means the corresponding brain network embeddings of normal people and disordered subjects also tend to be different. Therefore, it is desirable to consider the hubs when learning multi-view graph embeddings of brain networks, such that the resulted embeddings could better facilitate the group-contrasting analysis between brain disordered subjects and normal controls.

In Chapter 5, we propose to incorporate the hub detection task into the multi-view graph embedding framework so that the two tasks could benefit from each other. Specifically, we propose an auto-weighted framework of Multi-view Graph Embedding with Hub Detection (MVGE-HD) for brain network analysis. The MVGE-HD framework learns a unified graph embedding across all the views while reducing the potential influence of the hubs on blurring the boundaries between node clusters in the brain networks, thus leading to a clear and discriminative node clustering structure for the brain networks.

CHAPTER 2

SPATIO-TEMPORAL TENSOR ANALYSIS

(This chapter was previously published as "Spatio-Temporal Tensor Analysis for Whole-Brain fMRI Classification", in SDM'16 (Ma et al., 2016). DOI: https://doi.org/10.1137/1.9781611974348.92.)

2.1 Introduction

Many neurological disorders (*e.g.*, Alzheimer's disease (Ye et al., 2008), neuro-AIDS (Ragin et al., 2012b)) are characterized in the early stages by latent ongoing brain injury. As a forefront Neuroimaging technique, functional Magnetic Resonance Imaging (fMRI) has been widely used for noninvasive interrogation of the brain. During the course of an fMRI experiment, a series of brain images are obtained in the resting state or while the subject performs a task tailored to activate a specific cognitive function. Over the last decade, machine learning classifiers, especially kernel method, *e.g.*, Support Vector Machines (SVM), have been successfully employed on fMRI images for analysis of neurological status and diagnosis (Koch et al., 2012; Matthews et al., 2006).

In this work, we study the fMRI classification problem in the context of kernel modeling. Most work on fMRI classification focuses on analysis of specific brain regions of interest(ROI) (McKeown et al., 2007). However, ROI analysis is usually based on certain assumptions and may ignore additional valuable information in the image. Comparatively, whole-brain fMRI

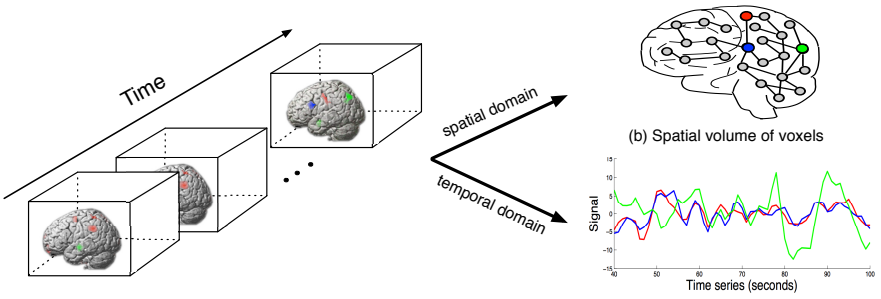


Figure 1: Example of fMRI brain images, which are inherently coupled with sophisticated spatio-temporal structure. Voxels are highly correlated with surrounding voxels in the spatial volume, and their signals are often very noisy in the time series.

images provide comprehensive structural and functional information of the human brain, thus having higher exploratory power and lower bias (Ecker et al., 2010; Song et al., 2015). Typically, as shown in Figure 1(a), a whole-brain fMRI image sample consists of a discrete time series of 3D image volumes (scans), where each volume consists of hundreds of thousands of voxels. Each voxel contains an intensity value that is proportional to the strength of the Nuclear Magnetic Resonance (NMR) signal emitted at the corresponding location in the brain volume (De Graaf and Koch, 2011). Therefore, an fMRI brain image sample can be naturally represented as a fourth-order tensor with 3D space $\times$ time. For example, if a fMRI brain scan acquired at a specific time point is a brain volume of dimensions $61 \times 73 \times 61$, then a sequence of 130 such scans collected at 130 separate time points would result in a discrete time series of length 130 for $61 \times 73 \times 61$ (*i.e.*, $271,633$) voxels, which can be represented as a $61 \times 73 \times 61 \times 130$ tensor. How to appropriately utilize the information from such sophisticated spatio-temporal structure is a main issue in the kernel modeling task.

Most of conventional kernel methods convert a tensor to a vector (or a matrix), which is then adapted in the kernel modeling (Signoretto et al., 2011; Zhao et al., 2013). However, voxels are often highly correlated with the surrounding voxels in the brain volume. For example, the adjacent voxels, marked with red and blue colors in Figure 1(b), exhibit similar patterns in Figure 1(c). This kind of tensor-to-vector (or tensor-to-matrix) conversion would cause the loss of structural information such as the spatial arrangement of voxel-based features, particularly given that fMRI data has high spatial resolution (Lindquist and others, 2008).

A common solution is to focus on the 3D spatial domain of the fMRI brain images (Song et al., 2015; He et al., 2014a). For instance, in (He et al., 2014a), the original 4D tensor of fMRI data is converted to a 3D tensor by averaging over the time dimension. Then the obtained 3D tensor is utilized in the kernel for classification. However, as shown in Figure 1(c), the signal in each voxel of the brain volume changes along with time. If the time series is averaged, the varying trend in the time series that reflects the brain activity will be lost. Some studies (Mørup et al., 2008; Deng et al., 2013) have focused on analyzing the fMRI time series of each individual voxel while ignoring structural information in the spatial domain. For instance, the multilinear decomposition model (Mørup et al., 2008) analyzes the time profile of the voxel vector converted from the 3D tensor in the spatial domain.

Although leveraging the spatio-temporal information is desired in building a predictive kernel method, it is very challenging due to the following three reasons:

Noisy fMRI time series analysis: Due to hardware reasons and subject factors (*e.g.*, thermal motion of electrons), there are often various nuisance components and random noise in

fMRI signals, leading to a low signal-to-noise ratio (SNR) (Lindquist and others, 2008). Since fMRI data has low temporal resolution, the signal of each voxel would not discriminatively change within a session of several time points, limiting ability to identify brain events in time frame. Furthermore, time shifts (delays), which occur naturally during the fMRI image acquisition process, should be taken into account while analyzing the data. How to filter the noise and extract discriminative information from the time series is critical in fMRI time series analysis.

Spatio-temporal feature extraction: Since fMRI data reflect brain activity from the spatial domain and temporal domain, a good feature extraction method should be able to extract a compact and informative representation from both domains while considering their correlations. Note that the time shift factor discussed previously should also be taken into consideration.

Kernel modeling: As discussed above, the existing works do not differentiate the spatial domain and temporal domain. How to incorporate both the correlation and the discrepancy between both domains into knowledge encoding is crucial for kernel modeling.

To deal with the above challenges, in this work, we propose a Spatio-Temporal Tensor Kernel (STTK) framework for whole-brain fMRI image analysis. Specifically, we first perform time series extraction to reduce the noise and filter out the less informative time points in the original volumetric time series. Then we utilize the shifted CANDECOMP/PARAFAC (SCP) (Harshman et al., 2003) factorization for feature extraction of the spatio-temporal data. Finally, spatio-temporal structure mapping is performed for kernel generation. Empirical studies on real-world resting-state fMRI brain images demonstrate that our proposed approach

can significantly boost the fMRI classification performance on divergent disease diagnosis (*i.e.*, Alzheimer's disease, ADHD and HIV).

2.2 Preliminaries

In this section we define some necessary notions and notations related to tensors and then present the problem formulation. Before proceeding, we introduce some basic notations that will be used throughout this work. Tensors (*i.e.*, multidimensional arrays) are denoted by calligraphic letters ($\mathcal{A}$, $\mathcal{B}$, $\mathcal{C}$, $\cdots$), matrices by boldface capital letters ($\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, $\cdots$), vectors by boldface lowercase letters ($\mathbf{a}$, $\mathbf{b}$, $\mathbf{c}$, $\cdots$), and scalars by lowercase letters (a, b, c, $\cdots$). The columns of a matrix are denoted by boldface lower letters with a subscript, *e.g.*, $\mathbf{a}_i$ is the ith column of matrix $\mathbf{A}$. The elements of a matrix or a tensor are denoted by lowercase letters with subscripts, *i.e.*, the $(i_1, \cdots, i_n)$ element of an n-th order tensor $\mathcal{A}$ is denoted by $a_{i_1,\ldots,i_n}$. $\mathbb{Z}^+$ is denoted by the set of positive integers. Additionally, we will often use Gothic letters ($\mathfrak{A}$, $\mathfrak{B}$, $\mathfrak{C}$, $\cdots$) to denote general sets or spaces, regardless of their specific nature.

2.2.1 Tensor Algebra

Definition 1 (Tensor)**.** An nth-order tensor is an element of the tensor product of n vector spaces, each of which has its own coordinate system.

Definition 2 (Tensor product)**.** Given order n and m tensors $\mathcal{A} \in \mathbb{R}^{I_1 \times \cdots \times I_n}$ and $\mathcal{B} \in \mathbb{R}^{I'_1 \times \cdots \times I'_m}$, their tensor product $\mathcal{A} \otimes \mathcal{B}$ is a tensor of order $n + m$ with the elements

$$(\mathcal{A} \otimes \mathcal{B})_{i_1,\ldots,i_n,i'_1,\ldots,i'_m} = a_{i_1,\cdots,i_n} b_{i'_1,\cdots,i'_m} \tag{2.1}$$

Note that a rank-one tensor of order n is the tensor product of n vectors. Clearly, an important operation applicable to our analysis is the tensor product (also called the outer product). The tensor product generalizes from the Kronecker product, but results in another tensor rather than a block matrix (Barnathan et al., 2010), which naturally endows tensor with the structure of tensor product representations and tensor product spaces. The space is equipped with inner product and norm.

Definition 3 (Inner product)**.** The inner product of two same-sized tensors $\mathcal{A}, \mathcal{B} \in \mathbb{R}^{I_1 \times \cdots \times I_n}$ is defined as the sum of the products of their elements:

$$\langle \mathcal{A}, \mathcal{B} \rangle = \sum_{i_1=1}^{I_1} \cdots \sum_{i_n=1}^{I_n} a_{i_1,\ldots,i_n} b_{i_1,\ldots,i_n} \tag{2.2}$$

Clearly, for rank-one tensors $\mathcal{A} = \mathbf{a}^{(1)} \otimes \cdots \otimes \mathbf{a}^{(n)}$ and $\mathcal{B} = \mathbf{b}^{(1)} \otimes \cdots \otimes \mathbf{b}^{(n)}$, it holds that

$$\langle \mathcal{A}, \mathcal{B} \rangle = \langle \mathbf{a}^{(1)}, \mathbf{b}^{(1)} \rangle \cdots \langle \mathbf{a}^{(n)}, \mathbf{b}^{(n)} \rangle \tag{2.3}$$

For brevity, we denote $\mathbf{x}^{(1)} \otimes \cdots \otimes \mathbf{x}^{(m)}$ by $\prod_{i=1}^{m} \otimes \mathbf{x}^{(i)}$.

Definition 4 (Norm)**.** The norm of a tensor $\mathcal{A}$ is defined to be the square root of the sum of all elements of the tensor squared, *i.e.*,

$$\|\mathcal{A}\| = \sqrt{\langle \mathcal{A}, \mathcal{A} \rangle} = \sqrt{\sum_{i_1=1}^{I_1} \cdots \sum_{i_n=1}^{I_n} a_{i_1,\ldots,i_n} a_{i_1,\ldots,i_n}} \tag{2.4}$$

2.2.2 Problem Formulation

In a typical fMRI classification task, we are given a collection of n training examples $\{\mathcal{X}_i, y_i\}_{i=1}^n \subset \mathfrak{X} \times \mathfrak{Y}$, where $\mathcal{X}_i \in \mathbb{R}^{I \times J \times K \times T}$ is the input fMRI sample with 3D space $\times$ time tensor form, and y_i is the class label of $\mathcal{X}_i$. The goal is to find a function $f : \mathfrak{X} \to \mathfrak{Y}$ that accurately predicts the label of an unseen example in $\mathfrak{X}$. In the kernel learning scenario, this problem can be formulated into the following optimization task:

$$f^* = \arg \min_{f \in \mathfrak{H}} \left(\frac{C}{n} \sum_{i=1}^{n} V\left(y_i, f(\mathcal{X}_i)\right) + \|f\|_{\mathfrak{H}}^2 \right), \tag{2.5}$$

where C controls the trade-off between the empirical risk and the regularization term $\|f\|_{\mathfrak{H}}^2$, $\mathfrak{H}$ is a set of functions forming a Hilbert space (the hypothesis space), and V is loss function that indicates how differences between y_i and $f(\mathcal{X}_i)$ should be penalized.

The attractiveness of kernel methods lies in its elegant treatment of nonlinear problems and its efficiency in high dimension. Different kernel methods or kernel machines arise from using different loss functions. In this work, we use the hinge loss function $\max\{0, 1 - y_i f(\mathcal{X}_i)\}$ for support vector machine (SVM).

2.3 Kernel Modeling

Two components of kernel methods need to be distinguished: the kernel machine and the kernel function. The kernel machine encapsulates the learning task, which usually can be formulated as an optimization problem (Jordanov and Jain, 2010). The kernel function encapsulates the hypothesis language, *i.e.*, how to perform data transformation and knowledge encoding. By

restricting to positive definite kernel functions, the optimization problem will be convex and solution will be unique. Throughout the work, we take "valid" to mean "positive definite".

Definition 5 (Positive Definite Kernel)**.** A symmetric function $\kappa : \mathfrak{X} \times \mathfrak{X} \to \mathbb{R}$ is a positive definite kernel on $\mathfrak{X}$ if, for all $n \in \mathbb{Z}^+$, $\mathcal{X}_1, \cdots, \mathcal{X}_n \in \mathfrak{X}$, and $c_1, \cdots, c_n \in \mathbb{R}$, it follows that $\sum_{i,j \in 1,\cdots,n} c_i c_j \kappa(\mathcal{X}_i, \mathcal{X}_j) \geq 0$.

A kernel function κ corresponds to the inner product in some feature space (a Hilbert space), which is in general different from the representation space of the instances. The computational attractiveness of kernel methods comes from the fact that quite often a closed form of 'feature space inner products' exists (Gärtner, 2003). Instead of mapping the data explicitly, the kernel can be calculated directly. According to *Mercer's theorem* (Vapnik, 2013), any valid kernel corresponds to an inner product in some feature space, and we can verify whether a kernel function is valid by the following Theorem (Berlinet and Thomas-Agnan, 2011).

Theorem 1. A function κ defined on : $\mathfrak{X} \times \mathfrak{X}$ is a positive definite kernel of $\mathfrak{H}$ if and only if there exists a feature mapping function $\phi : \mathfrak{X} \mapsto \mathfrak{H}$ such that

$$\kappa(\mathcal{X}, \mathcal{Y}) = \langle \phi(\mathcal{X}), \phi(\mathcal{Y}) \rangle \tag{2.6}$$

for any $(\mathcal{X}, \mathcal{Y}) \in \mathfrak{X} \times \mathfrak{X}$.

In particular, an important property of positive definite kernels is that they are closed under sum, multiplication by a scalar and product (Cristianini et al., 2000).

By the *representer theorem* (Schölkopf et al., 2001), the solutions of Equation 2.5 can be given by

$$f^*(\mathcal{X}) = \sum_{i=1}^{n} c_i \kappa(\mathcal{X}_i, \mathcal{X}), \tag{2.7}$$

where $c_i \in \mathbb{R}$ are suitable coefficients, and κ is a valid kernel of $\mathfrak{H}$.

2.4 Spatio-Temporal Tensor Kernel framework

From the above discussions, it is clear that a good kernel should be data dependent. As noted in the introduction, fMRI data are inherently coupled with spatio-temporal tensor structure, involving time shift and have very low temporal resolution and SNR. To facilitate kernel learning for fMRI data, we propose a spatio-temporal tensor kernel (STTK) framework that takes both the correlation and discrepancy between spatial and temporal domains into account. This framework consists of three steps: (1) volumetric time series extraction for extracting discriminative information from the time series, (2) spatio-temporal feature extraction for obtaining a more compact and informative representation, and (3) tensor structure mapping for kernel generation.

2.4.1 Volumetric Time Series Extraction

In fMRI time series extraction, a key issue is to determine the energy level for different time points. Most of existing work focus on single-voxel analysis (Mørup et al., 2008), while they ignore the spatial correlations between voxels, which may lead to suboptimal outcomes. In this section we develop a volumetric time series extraction approach for fMRI time series. In particular, we show how the volumetric (spatial) correlations and the temporal varying properties can contribute to the energy levels.

Given an fMRI example $\mathcal{X} \in \mathbb{R}^{I \times J \times K \times T}$, let $\mathbf{x}_{i,j,k}[t] = \{\mathbf{x}_{i,j,k}, t = 1, \cdots, T\}$ be a T-element time series of voxel $x_{i,j,k}$, and $\mathcal{X}[t]$ is a volume of $\mathcal{X}$ at time point t. $\{E(t, \mathcal{X}[t]), E(t, x_{i,j,k,t})\}$ is the energy function of time point t, where E is separated by volume and voxel for computational purposes and $E(t, \mathcal{X}[t]) = \{E_{min}(t, \mathcal{X}[t]), E_{max}(t, \mathcal{X}[t])\}$ correspond to the *minima* and the *maxima* to be defined later.

The choice of energy function plays a critical role in explaining how the knowledge transforms into meanings and contexts. The success of time series extraction strongly depends on the data knowledge encoded into the energy function. Two important points must be emphasized. First, in order to reduce the noise present in the measurement, new features should be used to describe voxels, rather than using the noisy voxel intensities as features. Second, due to the low temporal resolution, each voxel signal would not experience a discriminative changing within a short measurement time period. It is necessary to make a discriminant analysis along time prior to the volume measurements. Based on these two points, we propose the following three-step procedure:

Voxel Energy Measurement: We first extract the *maxima* and *minima* (*extrema*) points for each voxel's time series using the extrema extraction method (Fink and Gandhi, 2011), which is an effective and efficient technique for single-voxel time series extraction and noise removal (Deng et al., 2013). Let $\{(p_t, \mathbf{x}_{i,j,k}[p_t]), t = 1, \cdots, T_p\}$ and $\{(q_t, \mathbf{x}_{i,j,k}[q_t]), t = 1, \cdots, T_q\}$ be the *maxima* series and *minima* series of $\mathbf{x}_{i,j,k}[t]$, where p_t and q_t are the time indexes, T_p and T_q are the number of *maxima* and *minma*, repsectively. Then, for each voxel $x_{i,j,k,t}$, we measure its energy by

$$E(t, x_{i,j,k,t}) = \begin{cases} 1, & \text{if } t \in p_t \\ -1, & \text{if } t \in q_t \\ 0, & \text{otherwise} \end{cases} \tag{2.8}$$

where the values of 1 and -1 mean 'importance', and 0 means 'no importance'.

Volumetric Energy Measurement: We measure the energy of each volume by summing up the energies of all the voxels in it. In particular, we separately consider the maxima and minima voxels by

$$E_{max}(t, \mathcal{X}[t]) = \sum_{i,j,k} \max(E(t, x_{i,j,k,t}), 0) \tag{2.9}$$

$$E_{min}(t, \mathcal{X}[t]) = \sum_{i,j,k} \max(-E(t, x_{i,j,k,t}), 0) \tag{2.10}$$

Volumetric Time Series Extraction: We extract the time series from measured volumes based on E_{max} and E_{min}. Let ER_t be the time series extraction rate defined by N/T, where N is the number of extracted time points. Given an extraction rate ER_t, we first rank all the volumetric time points according to E_{max} and E_{min} respectively. Then we select the top-k time points from each of the two ranked time point sets and concatenate them, which forms the extracted time series, where k equals to $ER_t \times T/2$.

As an illustration, Figure 2 shows the time series of a voxel with different time series extraction techniques. From the original time series (a), we can see that it is nontrivial to distinguish activation fluctuations from the background noise if no time series extraction is

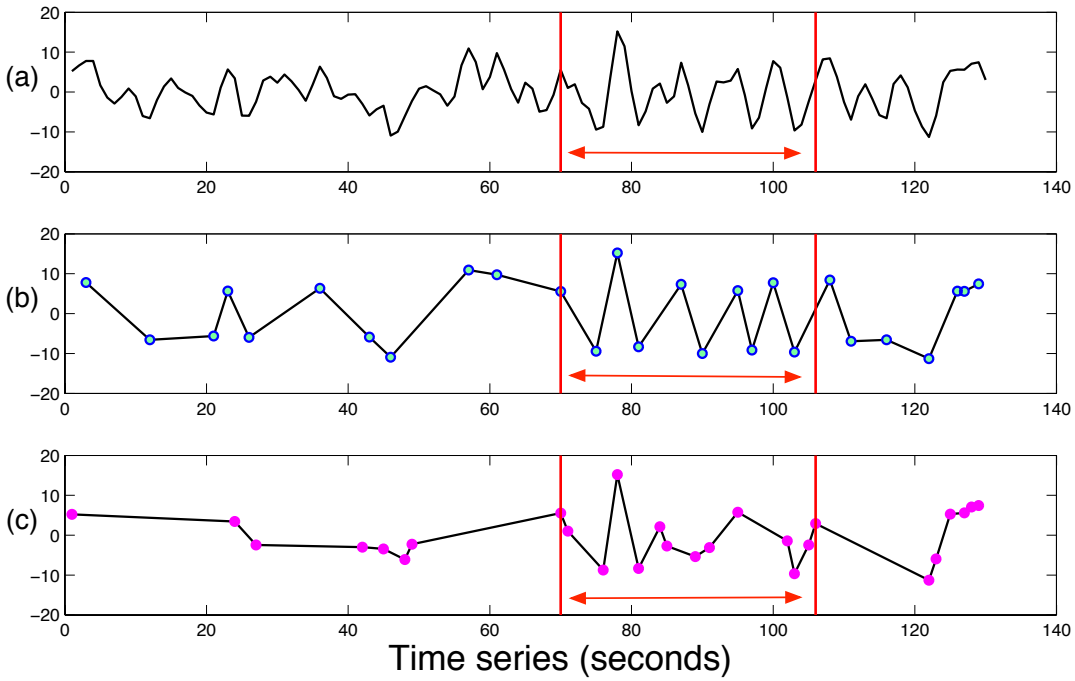
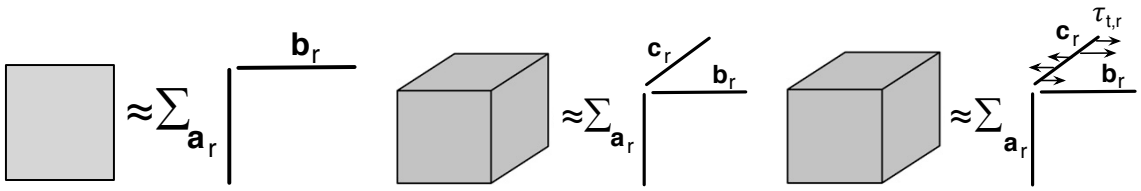


Figure 2: Illustration of the time series of a voxel with different time series extraction techniques. Each circle stands for an extracted time point. (a) is the original time series, (b) is the sequence extracted using single-voxel technique, where the time points with top 20% absolute values are extracted, and (c) is the sequence extracted using our approach, with $ER_t = 0.2$. Significant changes of signal occur in the interval between red lines.

performed. Comparing with the time series (b) extracted using single-voxel technique, the time series (c), with the same amount of sampling time points as (b), extracted by our volumetric approach can better capture the significant changes of signal over time. For example, during the time interval $[70, 105]$ (between red lines), the signals in (c) experience notable irregular changes, which can also be observed from (a). Comparatively, (b) only captures the most distinct changes within this period. For the period $[0, 70]$, the original series shows slightly



(a) Matrix Factorization (b) CP Factorization (c) Shifted CP Factorization

Figure 3: CP factorization is a generalization of matrix factorization to tensors. The SCP model allows shifts to occur over the second mode such that for each index of the third mode the component of the second mode is shifted a given amount.

fluctuated changes, which can also be reflected by (c), while the time series (b) has much more changes.

This is majorly because the single-voxel technique chooses time points only based on the extrema of the single-voxel time series. In contrast, our approach performs the extraction based on the time varying volume series. Since the voxels of different regions in human brain are highly correlated and they usually collaboratively participate in a brain activity, their overall changing trend could better reflect the brain activity. By considering the time series of all the voxels in the volume, our extraction method incorporates both the spatial correlation of the volumetric voxels and the varying properties in the temporal domain into the analysis. Therefore, it can bring us more discriminative time series for fMRI brain image analysis.

2.4.2 Spatio-Temporal Feature Extraction

Tensors provide a natural representation for fMRI data, but there is no guarantee that such representation will be good for kernel learning. From the characteristics of tensor, we know that the essential information in the tensor is embedded in its multi-way structure. Thus,

one important aspect of kernel learning for such complex objects is to represent them by sets of key structural features which are easier to manipulate. Most of the previous work use CANDECOMP/PARAFAC (CP) factorization (as shown in Figure 3) for fMRI data analysis, but it cannot well capture the structural information of a spatio-temporal tensor. Recently, it was found that shifted CP (SCP) factorization (Mørup et al., 2008) is particularly effective for extracting such spatio-temporal structure. It can simultaneously consider the inter-mode correlations and the time shift in fMRI data, yielding a more compact representation of fMRI data. Motivated by these observations, we utilize SCP factorization to further perform feature extraction.

Given a tensor $\mathcal{X} \in \mathbb{R}^{I \times J \times K \times T}$, SCP factorizes it as

$$\mathcal{X} = \sum_{r=1}^{R} \mathbf{a}_r \otimes \mathbf{b}_r \otimes \mathbf{c}_r \otimes \mathbf{d}_r^{\tau} + \mathcal{E}, \tag{2.11}$$

where R is the rank of the tensor $\mathcal{X}$ defined as the smallest number of rank-one tensors in an exact SCP factorization, and the superscript τ denotes that the time shift will be along the forth mode (see Figure 3 for an example), and $\mathcal{E}$ is the residual.

Remark that although SCP factorizes the data tensor, we can still recover the original data from the factorized results.

2.4.3 Tensor Structure Mapping

Let us now consider how the above feature extraction results can be exploited to induce a kernel. Suppose we are given the SCP factorization of $\mathcal{X}, \mathcal{Y} \in \mathbb{R}^{I \times J \times K \times T}$ by $\mathcal{X} = \sum_{r=1}^{R} \mathbf{x}_r^{(1)} \otimes$

$\mathbf{x}_r^{(2)} \otimes \mathbf{x}_r^{(3)} \otimes \mathbf{x}_r^{(4)\tau}$ and $\mathcal{Y} = \sum_{r=1}^{R} \mathbf{y}_r^{(1)} \otimes \mathbf{y}_r^{(2)} \otimes \mathbf{y}_r^{(3)} \otimes \mathbf{y}_r^{(4)\tau}$ respectively. We assume the tensor observations are mapped into the Hilbert space $\mathfrak{H}$ by

$$\phi : \mathcal{X} \to \phi(\mathcal{X}) \in \mathbb{R}^{H_1 \times H_2 \times H_3 \times H_4^{\tau}}. \tag{2.12}$$

Note that the projected tensor $\phi(\mathcal{X})$ has the same order with $\mathcal{X}$, but each mode dimension is higher and it is even an infinite dimension depending on the feature mapping function $\phi(.)$.

Based on the definition of the kernel function, it is easy to find that the feature space is a high-dimensional space of the original space, equipped with the same operations. Thus, we can factorize tensor data directly in the feature space in the same way as in the original space. This is formally equivalent to performing the following mapping:

$$\phi : \sum_{r=1}^{R} \prod_{i=1}^{3} \otimes \mathbf{x}_r^{(i)} \otimes \mathbf{x}_r^{(4)\tau} \to \sum_{r=1}^{R} \prod_{i=1}^{3} \otimes \phi(\mathbf{x}_r^{(i)}) \otimes \phi(\mathbf{x}_r^{(4)\tau}). \tag{2.13}$$

In this sense, it corresponds to mapping tensors into high-dimensional tensors that retain the original structure. More precisely, it can be regarded as mapping the original data into tensor feature space and then conducting the SCP factorization in the feature space.

After mapping the SCP factorization of the data into the tensor product feature space, the kernel can be defined directly with the inner product in that feature space. Thus, we derive our STTK:

$$\kappa\left(\sum_{r=1}^{R}\prod_{i=1}^{3}\otimes\mathbf{x}_r^{(i)}\otimes\mathbf{x}_r^{(4)\tau},\sum_{r=1}^{R}\prod_{i=1}^{3}\otimes\mathbf{y}_r^{(i)}\otimes\mathbf{y}_r^{(4)\tau}\right)=\sum_{p=1}^{R}\sum_{q=1}^{R}\prod_{i=1}^{3}\kappa\left(\mathbf{x}_p^{(i)},\mathbf{y}_q^{(i)}\right)\kappa\left(\mathbf{x}_p^{(4)\tau},\mathbf{y}_q^{(4)\tau}\right). \tag{2.14}$$

From its derivation, we know such a kernel can take the multi-way spatio-temporal structure flexibility into account. In general, the STTK is an extension of the conventional kernels in the vector space to tensor space, and each vector kernel can be used in this framework for fMRI classification analysis in conjunction with kernel machines. Our positive result can be viewed as that designing a good tensor kernel function tends to be equivalent to designing a good tensor structure in the feature space (Balcan et al., 2006).

2.5 Experiments and Evaluation

In order to empirically evaluate the effectiveness of the proposed approach for fMRI classification, we test our model on real fMRI data and compare with several state-of-the-art kernel methods in fMRI study.

2.5.1 Data Collection and Preprocessing

In this work, we consider three real resting-state whole-brain fMRI image datasets as follows:

- *Alzheimer's Disease (ADNI)*[1]: It contains fMRI images of 33 subjects, each with a series of $61 \times 73 \times 61$ scans for 130 time points. These subjects are AD patients (positive) or normal people (negative).

[1] http://adni.loni.usc.edu/

- *Human Immunodeficiency Virus Infection (HIV)* (Wang et al., 2011a): This dataset contains fMRI brain images of 83 subjects, each with a series of $61 \times 73 \times 61$ scans for 255 time points. These subjects are early HIV patients (positive) or normal controls (negative).
- *Attention Deficit Hyperactivity Disorder (ADHD)*[1]: This dataset contains the resting-state fMRI images of 100 subjects, each with a series of scans for $58 \times 49 \times 47$ voxels. Subjects are either ADHD patients (positive) or normal controls (negative). Different from previous datasets, the lengths of time series for different subjects in ADHD dataset are not the same, ranging from 74 to 257.

In the derived datasets, each 3D fMRI scan has the NIFTI format. We convert each scan to a 3D tensor using SPM8[2]. Then we use SPM8 toolbox to preprocess these data, including images realignment, slice timing correction and normalization. We also perform spatial smoothing on these functional images with an 8mm FWHM Gaussian kernel for increasing signal-to-noise ratio (SNR). REST[3] is used afterwards for band-pass filtering (0.01-0.08 Hz) and linear trend removing of the time series.

2.5.2 Baselines and Metrics

In order to establish a comparative study, we use five kernel learning methods as baselines. We use the classification accuracy as the evaluation metric.

---

[1]http://neurobureau.projects.nitrc.org/ADHD200/

[2]http://www.l.ion.uc.ac.uk/spm/software/spm8

[3]http://resting-fmri.sourceforge.net

TABLE I: Summary of compared methods. ST means Spatio-Temporal, C is the trade-off parameter, σ is the kernel width parameter, R is the rank of tensor factorization, and ER_t is the time series extraction rate.

Property	FK	sKL	S-DuSK	ST-DuSK	STTK$_{nonTE}$	STTK
Type of Input Data	Unfolded Matrices	Unfolded Matrices	Spatial Tensor	ST Tensor	ST Tensor	ST Tensor
Type of ST Correlation Exploited	One-way	One-way	Three-way	Multi-way	Multi-way	Multi-way
Differentiating Space v.s. Time	No	No	No	No	Yes	Yes
Time Series Feature Extraction	No	No	No	No	No	Yes
Parameters	C, σ	C, σ, R	C, σ, R	C, σ, R	C, σ, R	C, σ, R, ER_t

- **Factor kernel (FK)** (Signoretto et al., 2011): a matrix unfolding based tensor kernel. The constituent kernels are from the class of Gaussian RBF kernels.
- **sKL** (Zhao et al., 2013): a kernel defined based on the symmetric Kullback-Leibler divergence, where the tensors are also unfolded into matrices, which has been applied to reconstruct 3D movement.
- **DuSK** (He et al., 2014a): a tensor kernel based upon CP factorization. The authors average the fMRI data over the temporal dimension and apply DuSK on the obtained 3D fMRI data. For evaluation, we implement DuSK in both the 3D spatial data setting and the 4D spatio-temporal data setting, which are denoted as **S-DuSK** and **ST-DuSK**, respectively.
- **STTK:** our proposed spatio-temporal tensor kernel. To evaluate the effectiveness of volumetric time series extraction, we employ STTK with and without volumetric time series extraction and denote them as **STTK** and **STTK**$_{nonTE}$ respectively. Specifically, to study the importance of temporal correlations of the fMRI brain images within the time series, we randomly permute the order of the time dimension of the fMRI data, and then apply our STTK to it. We denote this case as **STTK**$_{permT}$.

We apply each kernel learning method in SVM and evaluate their performance. Specifically, we apply all the six methods on ADNI and HIV datasets. For the ADHD dataset, the lengths of the time series are different for different subjects, while Factor kernel, sKL, ST-DuSK and STTK_{nonTE} require dimensions of different samples must agree. Thus, we only apply S-DuSK, STTK_{permT} and STTK on ADHD dataset. We use LibSVM (Chang and Lin, 2011), a widely used implementation of SVM, with Gaussian RBF kernel as the classifier. Table I summarized the compared methods. The optimal trade-off parameter for all the methods is selected from $C \in \{2^{-5}, 2^{-4}, \ldots, 2^5\}$, the kernel width parameter is selected from $\sigma \in \{2^{-5}, 2^{-4}, \ldots, 2^5\}$, the optimal rank R is determined by grid search from $\{1, 2, \ldots, 8\}$, and the time series extraction rate ER_t is chosen from $[0, 1]$. Here we set the time series extraction rate ER_t to be 0.2, *i.e.*, only 20% of the time sequences will be kept. In the experiment, 5-fold cross validations are performed. We repeated this process for 50 times and report the average classification accuracy as the result.

2.5.3 Classification Performance

As shown in Table II, our STTK method performs the best on all three datasets in terms of classification accuracy. Among the listed kernel methods, Factor kernel and sKL unfold the original tensor data into matrices while all the other methods preserve the spatial tensor structure during the learning process. As can be seen from the results, Factor kernel and sKL achieve a relatively lower accuracy on both the ADNI dataset and HIV dataset. This implies that unfolding tensor into matrices would lose the spatial structural information, leading to the degraded performance. Another observation is that DuSK achieves a quite high accuracy

when applied in the three-dimensional spatial data setting, while the accuracy decreases to a great extent on the four-dimensional spatio-temporal fMRI data. This is majorly due to the fact that the time series of fMRI data are very noisy, involving time shift and with low SNR. Extending DuSK to the spatio-temporal domain without proper treatments would even damage its performance.

Comparatively, our proposed STTK properly encodes the prior knowledge of time series analysis with the spatio- temporal structural information into one tensor based kernel model. Therefore, the classification accuracy of STTK is much higher than that of ST-DuSK, especially on the ADNI dataset. Furthermore, by extracting the most significant features in the time series at an appropriate compression rate, our STTK can better discriminate the fMRI patterns with different medical status. Meanwhile, this volumetric time series extraction strategy enables us to analyze fMRI time series with different lengths (*e.g.*, the ADHD dataset used in the experiment). The experimental results demonstrate the effectiveness and considerable advantages of our proposed methods in the fMRI study.

As can be seen in Table II, another notable result is that STTK_{permT} achieves a much lower accuracy than STTK, which means the random permutation of the temporal sequential order of fMRI brain images degrades the classification performance. This implies that the temporal order of the fMRI brain images is very important for the classification. This is mainly because that the original varying trend of the fMRI time series reflects the sequential brain activity within the period. If the temporal order of the fMRI brain images is permuted, the original

TABLE II: Classification accuracy comparison (mean ± standard deviation)

	ADNI	HIV	ADHD
FK	0.593 ± 0.029	0.663 ± 0.011	N/A
sKL	0.510 ± 0.030	0.645 ± 0.021	N/A
S-DuSK	0.731 ± 0.021	0.718 ± 0.005	0.622 ± 0.010
ST-DuSK	0.576 ± 0.052	0.642 ± 0.023	N/A
STTK$_{nonTE}$	0.710 ± 0.010	0.693 ± 0.006	N/A
STTK$_{permT}$	0.583 ± 0.020	0.615 ± 0.021	0.594 ± 0.018
STTK	$\mathbf{0.759 \pm 0.022}$	$\mathbf{0.762 \pm 0.010}$	$\mathbf{0.680 \pm 0.013}$

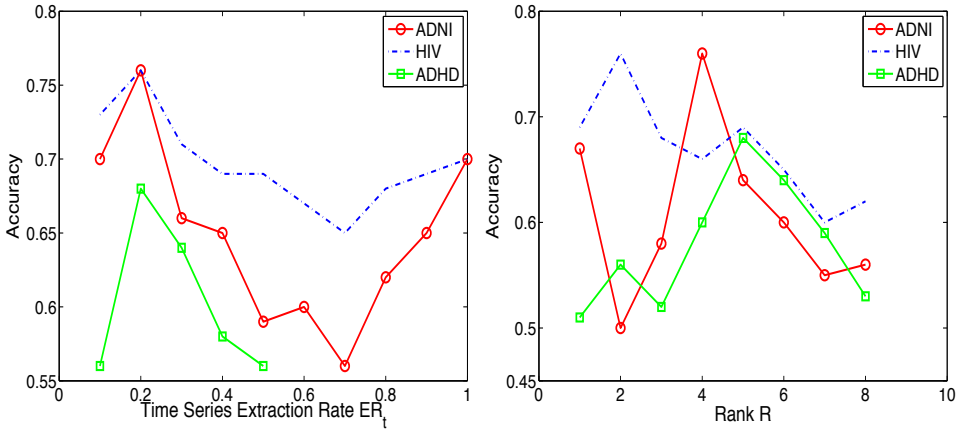


Figure 4: Parameter sensitivity

temporal correlation of the fMRI brain images would be damaged. This result also demonstrates that our STTK method captures the temporal correlation of fMRI data well.

2.5.4 Parameter Sensitivity

Although the optimal values of the parameters in our proposed STTK are found using cross-validation, it is of interest to see the sensitivity of STTK to the time series extraction rate ER_t and the rank of tensor factorization R.

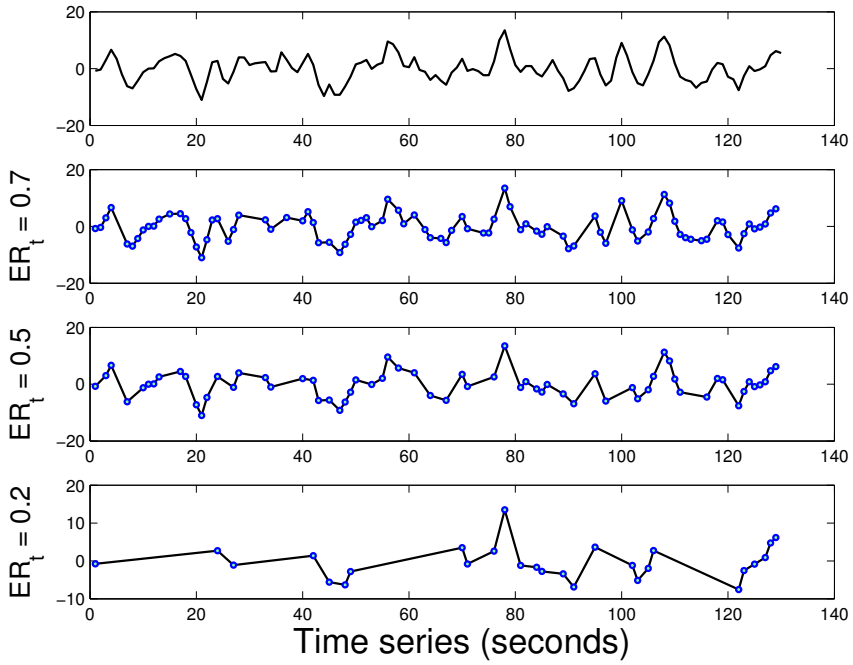


Figure 5: Time series of a voxel extracted with varying time series extraction rate ER_t.

We first evaluate the classification performance of STTK with varying ER_t. We vary ER_t from 0.1 to 1.0 on ADNI and HIV datasets. For ADHD dataset, the lengths of time series for different subjects are quite different, varying from 74 to 257. We extract the same number of time points from each of them, and then compute the average extraction rate, and use it as the extraction rate for ADHD dataset. Since the average extraction rate reaches its maximum around 0.58 due to the different lengths of the time series, here we vary ER_t from 0.1 to 0.5 for the evaluation on ADHD dataset. As shown in Figure 4, the value of ER_t significantly impacts the classification accuracy. We can find that the accuracy declines when

$ER_t > 0.2$. This indicates, counterintuitively, keeping more time points (with higher ER_t) does not improve the accuracy; instead, it may even lead to a worse performance. As illustrated in Figure 5, the time series extracted with $ER_t = 0.5$ and the one extracted with $ER_t = 0.7$ contain many redundant time points, especially in the time interval [0, 70], which may degrade the performance. Although keeping even more time points might be helpful, as the accuracy starts to increase when $ER_t > 0.7$, we can notice that the optimal results for all datasets are achieved when $ER_t = 0.2$. This reflects the fact that fMRI time series are commonly noisy, containing many redundant time points that are insignificant for disease diagnosis. With an appropriate value of ER_t, the volumetric time series extraction enables STTK to greatly filter the background noise, while preserving the most discriminative patterns in the fMRI time series.

Next, we evaluate the sensitivity of STTK to the rank R of tensor factorization. We fix ER_t at 0.2 which is the optimal value for each dataset, and vary R from 1 to 8 with a step size of 1. As shown in Figure 4, the rank parameter R has a significant effect on the classification accuracy and the optimal value of R depends on the datasets. In general, the optimal value of R lies in the range between 2 and 5, which may provide a good guidance for selection of the R in advance. How to determine the optimal rank for a specific tensor factorization method is beyond the scope of this work and still remains an open research problem (De Silva and Lim, 2008; Hao et al., 2013).

2.6 Related Work

Our work relates to a vast literature on spatio-temporal data analysis, tensor analysis techniques, and kernel learning. We present a selection of such works below.

Spatio-Temporal Data Analysis: Spatio-temporal data analysis has attracted considerable attention recently. Many models have been conducted to address the challenges in different contexts (Cressie and Wikle, 2015). However, these models usually require domain knowledge since they make strong assumptions on the spatial and temporal correlation of the data. Some models have been used in the spatio-temporal fMRI brain image analysis (Oikonomou et al., 2012), while most of them treat spatial domain and temporal domain separately. For instance, in (Haller et al., 2007), spatial analysis is performed via general linear modelling (GLM), while temporal analysis is done with a direct comparison of BOLD response estimates between regions.

Tensor Factorizations: Our work is also motivated by recent advances in tensor factorization and its applications in the fMRI data analysis (Kuang et al., 2013). A comprehensive survey on tensor factorization can be found in (Kolda and Bader, 2009). One of the most commonly used one is CP factorization. In the spatio-temporal tensor setting, the shifted CP is more frequently used (Mørup et al., 2008), but for exploratory analysis. In this study, we employ it to facilitate kernel learning.

Kernel learning: Several tensor based kernel methods have been recently investigated (Signoretto et al., 2011; Zhao et al., 2013; Gärtner, 2003). Most of them focus on learning kernel via matrix unfolding, thus only capturing the one-way relationship within the tensor data. The multi-way structures within tensor data are already lost before the kernel construction. The problem of how to build kernel directly on tensor data has not been well studied. A first attempt in this direction is related to CP factorization proposed in (He et al., 2014a), while it has the same drawback as CP factorization.

CHAPTER 3

MULTI-GRAPH CLUSTERING

(This chapter was previously published as "Multi-Graph Clustering based on Interior-Node Topology with Applications to Brain Networks", in ECML/PKDD '16 (Ma et al., 2016).DOI: https://doi.org/10.1007/978-3-319-46128-1_30.)

3.1 Introduction

In recent years, graph mining has been a popular research area because of numerous applications in social network analysis, computational biology and computer networking. In addition, many new kinds of data can be represented as graphs. For example, from common brain images such as the functional magnetic resonance imaging (fMRI) data of multiple subjects, we can construct a brain connectivity network for each of them, where each node represents a brain region, and each link represents the functional/structural connectivity between two brain regions (Kong and Yu, 2014). These multiple brain networks provide us with an unprecedented opportunity to explore the inner structure and activity of the human brain, serving as valuable supportive information for clinical diagnosis of neurological disorders (Ragin et al., 2012a). Therefore, mining on graphs becomes a crucial task and may benefit various real-world applications.

Among the existing works on graph learning, quite a few of them fall into supervised learning, which usually aim to select frequent substructures such as connected subgraph patterns in a

database of graphs and then feed these subgraph features into classifiers (Cao et al., 2015b; Kong et al., 2013). These methods typically work well when the graph database is very large or the access to side information is assumed. However, the number of subgraphs is exponential to the size of graphs, thus the subgraph enumeration process is both time and memory consuming which makes it infeasible to explore the complete subgraph space. Moreover, in many real-world cases, only a small number of labeled graphs are available. Therefore, finding discriminative subgraph patterns from a large number of candidate patterns based on such limited instances is not reliable. While supervised methods focus on training classifiers, unsupervised clustering could provide exploratory techniques for finding hidden patterns in multiple graphs. In this work, we investigate the unsupervised scenarios by exploring the multi-graph clustering based on the interior-node topology of graphs. Topology is the mathematics of neighborhood relationships in space, which is independent of the distance metric, thus the interior-node topology of graphs could provide complementary local structure information for the origninal linkage, which can only characterize the global structure information of graph. Despite its value and significance, to our best knowledge, the interior-node topology of graphs has not been studied in the problem of multi-graph clustering so far. There are two major challenges in this multi-graph clustering problem:

- How to capture the interior-node topology of each graph? Conventional approaches extract graph-theoretical measures, *e.g.*, clustering coefficients, to quantify the prevalence of clustered connectivity (Jie et al., 2014; Wee et al., 2012). However, assigning a predefined measure to

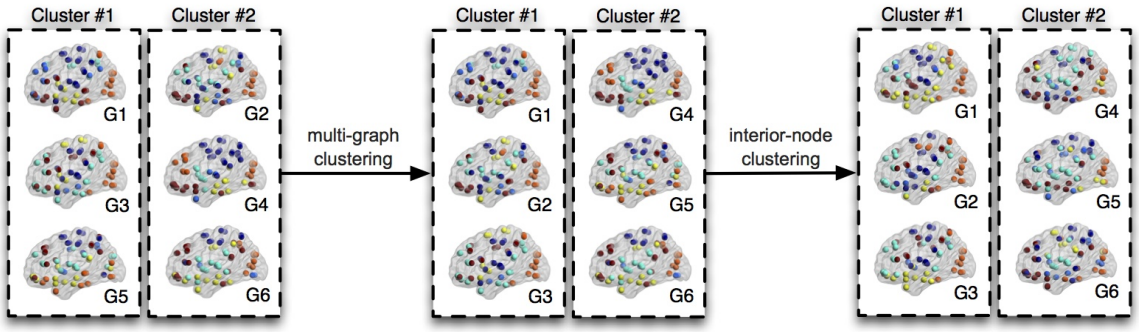


Figure 6: The framework of the proposed model.

specific nodes in a graph might not fully characterize the subtle local topological structure of the graph.

- How to effectively leverage the extracted topological structure information to facilitate the process of multi-graph clustering? The original linkage metric describes the global connectivity structure in the graph, while the topological structure depicts the local neighborhood relationships. An effective multi-graph clustering model should fuse these two complementary structural information together.

To address the above challenges, we propose a framework of multi-graph clustering with interior-node topology. The contributions of this work are twofold:

- We propose to consider both the global structure and the local topological structure of graphs for the multi-graph clustering task. Specifically, we utilize interior-node clustering to capture local topological structure of graphs.

- Considering the fact that graphs with a high similarity tend to have a similar interior-node topology, we propose to merge the multi-graph clustering stage and interior-node clustering process into a unified iterative framework called MGCT, where the results of interior-node clustering are exerted on multi-graph clustering and the multi-graph clustering will in turn improve interior-node clustering of each graph, thus achieving a mutual reinforcement.

In the scenario of brain network analysis for multiple subjects, the proposed framework of multi-graph clustering can be illustrated with the example shown in Figure 6. There are two stages in each iteration of the framework: multi-graph clustering and interior-node clustering. In the multi-graph clustering stage, the given six brain networks are clustered into two clusters, and then in the second stage, the interior-node clustering of each graph will be updated with a weighted influence from their neighbor graphs in the same cluster, after which the new interior-node clustering results will be utilized for the multi-graph clustering in the next iteration. After the model converges, we will obtain the final optimal multi-graph clustering results, which can be used for further analysis, for example, the neurological disorder identification.

We evaluate the proposed method on two real brain network data sets (ADHD and HIV). Experimental results illustrate the superior performance of the proposed approach for multi-graph clustering in brain network analysis.

3.2 Preliminaries

In this section we establish key definitions and notational conventions that simplify the exposition in later sections.

Throughout this work, matrices are written as boldface capital letters and vectors are denoted as boldface lowercase letters. For a matrix $\mathbf{M} \in \mathbb{R}^{n \times m}$, its elements are denoted by m_{ij}, and its i-th row, j-th column are denoted by $\mathbf{m}^i$, $\mathbf{m}_j$ respectively. The Frobenius norm of $\mathbf{M}$ is defined as $\|\mathbf{M}\|_F = \sqrt{\sum_{i=1}^{n} \|\mathbf{m}^i\|_2^2}$, and the $\ell_{2,1}$ norm of $\mathbf{M}$ is defined as $\|\mathbf{M}\|_{2,1} = \sum_{i=1}^{n} \left\|\mathbf{m}^i\right\|_2$. For any vector $\mathbf{u} \in \mathbb{R}^n$, $Diag(\mathbf{u}) \in \mathbb{R}^{n \times n}$ is the diagonal matrix whose diagonal elements are u_i. $\mathbf{I}_n$ denotes an identity matrix with size n. $\|\mathbf{u}\|_0$ is the ℓ_0 norm, which counts the number of nonzero elements in the vector $\mathbf{u}$.

Definition 1 (Multi-graph Clustering) An undirected graph can be formally represented as $G = (V, E, \mathbf{A})$, where V is the set of vertices, $E \subset V \times V$ is the set of edges, and $\mathbf{A}$ is the weighted affinity matrix whose entry denotes the affinity between a pair of nodes. Given a set of such graphs $D = \{G_1, G_2, \cdots, G_n\}$, the goal of multi-graph clustering is to cluster the graphs in D into c subsets.

Definition 2 (Interior-node Clustering) Given an undirected graph $G = (V, E, \mathbf{A})$, the goal of interior-node clustering is to group the nodes of the graph into k clusters $C = \{C_1, \cdots, C_k\}$, with $V = C_1 \cup \cdots \cup C_k$ and $C_i \cap C_j = \varnothing$ for every pair i, j with $i \neq j$.

Definition 3 (Topology) Topology is the mathematics of neighborhood relationships in space independent of metric distance. In the context of graph structures, such neighborhood relationships often correspond to the *connectivity* of nodes, *i.e.,* how nodes are connected to each other (King, 2002).

3.3 Methodology

In this section, we first introduce the proposed multi-graph clustering framework MGCT, where we formulate the multi-graph clustering stage and the interior-node clustering stage, both of which can be formulated as optimization problems. We then present an iterative algorithm based on half-quadratic optimization to solve this minimization problem.

3.3.1 An Iterative Framework: MGCT

In the literature of multi-graph clustering, the pairwise distance is mainly measured based on the structure of each graph, and graphs with highly similar structures tend to be clustered into the same group. In other words, the graphs that are clustered into the same group tend to have highly similar topological structure (Aggarwal and Wang, 2010). Following these observations, we propose an iterative framework called MGCT for multiple-graph clustering based on interior-node topology. In each iteration, there are two stages: the interior-node clustering and the multi-graph clustering, where the interior-node clustering results which imply local topological structure are used together with the global structure of graph for clustering multiple graphs, and then the multi-graph clustering results will be utilized in turn to improve the interior-node clustering. Through this iterative mutual reinforcement of interior-node clustering and multi-graph clustering, we can finally achieve a refined multi-graph clustering result.

Multi-Graph Clustering In this part, we focus on the formulation of the multi-graph clustering stage. Since the multi-graph clustering and interior-node clustering depend on each other and are performed alternatively, here we assume we have obtained the interior-node clustering

results of the graphs, which can be used for the multi-graph clustering. The formulation of the interior-node clustering problem and the overall iterative process will be discussed later.

Given a set of graphs $D = \{G_1, G_2, \cdots, G_n\}$, with the corresponding set of affinity matrices $A = \{\mathbf{A}_1, \mathbf{A}_2, \cdots, \mathbf{A}_n\}$, where $\mathbf{A}_i \in \mathbb{R}^{m \times m}$ is the weighted affinity matrix of G_i, and its entry denotes the pairwise affinity between nodes in G_i, suppose we have performed interior-node clustering on each of these graphs and obtained a set of clustering indicators $F = \{\mathbf{F}_1, \mathbf{F}_2, \cdots, \mathbf{F}_n\}$, where $\mathbf{F}_i \in \mathbb{R}^{m \times k}$ is the interior-node clustering indicator of G_i, we build a similarity matrix $\mathbf{S} \in \mathbb{R}^{n \times n}$, where s_{ij} denotes the similarity between the two graphs G_i and G_j, and we define it as:

$$s_{ij} = \delta(- \|\mathbf{A}_i - \mathbf{A}_j\|_F^2) + (1 - \delta)(- \|\mathbf{F}_i - \mathbf{F}_j\|_F^2) \tag{3.1}$$

which is a weighted combination of the similarity based on the original affinity matrix of each graph and the similarity based on interior-node clustering results, where δ is the weight parameter balancing the two parts. In this way, the interior-node topology characterized by the interior-node clustering indicator matrix can be incorporated for multi-graph clustering. With this similarity matrix, we can formulate the clustering of graphs in D as a spectral clustering problem, where graphs with a higher pairwise similarity tend to be grouped into the same clus-

ter. Let $\mathbf{H} \in \mathbb{R}^{n \times c}$ be the multi-graph clustering indicator matrix, then the optimal $\mathbf{H}$ can be obtained by solving the following objective function (Von Luxburg, 2007):

$$\begin{aligned} &\min_{\mathbf{H}} \operatorname{Tr}\left(\mathbf{H}^{\mathrm{T}} \mathbf{L} \mathbf{H}\right) \\ &\text{s.t.} \quad \mathbf{H}^{\mathrm{T}} \mathbf{H} = \mathbf{I}_c \end{aligned} \tag{3.2}$$

where $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{S})\mathbf{D}^{-\frac{1}{2}}$ is the symmetric normalized Laplacian matrix, and $\mathbf{D}$ is a diagonal matrix with $d_{ii} = \sum_{j=1}^{n} s_{ij}$.

Interior-Node Clustering We now study the problem of interior-node clustering of graph in the context of multi-graph clustering.

In graph theory, a cluster is described as a set of nodes more densely connected with each other than with the rest nodes of the graph. Given a graph G with m nodes and the weighted affinity matrix $\mathbf{A} \in \mathbb{R}^{m \times m}$, the goal of interior-node clustering is to group the m nodes into k clusters, *i.e.,* to find a cluster indicator matrix $\mathbf{F} \in \mathbb{R}^{m \times k}$, whose entry indicates which cluster a node may belong to.

Intuitively, nodes with a higher correlation should have a similar cluster indicator. With this assumption, a graph regularization can be embedded to learn the cluster indicator matrix $\mathbf{F}$,

which is formulated as the following minimization problem on the basis of the spectral analysis (Von Luxburg, 2007):

$$\min_{\mathbf{F}} \sum_{i,j=1}^{m} a_{ij} \left\| \frac{\mathbf{f}^i}{\sqrt{d_{ii}}} - \frac{\mathbf{f}^j}{\sqrt{d_{jj}}} \right\|_2^2 = \mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}} \mathbf{L}' \mathbf{F}\right)$$
$$\text{s.t.} \quad \mathbf{F}^{\mathrm{T}} \mathbf{F} = \mathbf{I}_k \tag{3.3}$$

where $\mathbf{L}' = \mathbf{D}'^{-\frac{1}{2}}(\mathbf{D}' - \mathbf{A})\mathbf{D}'^{-\frac{1}{2}}$ is the symmetric normalized Laplacian matrix, and $\mathbf{D}'$ is a diagonal matrix with $d_{ii} = \sum_{j=1}^{m} a_{ij}$.

The above formulation provides a measure of the smoothness of $\mathbf{F}$ over the edges in G. Notice that when a node connects to the nodes in different clusters, it will lead to a relatively large value of $\mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}} \mathbf{L}' \mathbf{F}\right)$ (Spielman, 2010). Therefore, it is expected to identify these boundary-spanning nodes to moderate this influence. In the following, we show how to model and leverage the topology of interior-node to achieve this goal.

From the definition of the topology, we know it is the mathematics of neighborhood relationships in space independent of metric distance. In the context of graph structures, such neighborhood relationships often correspond to the *connectivity* of nodes, *i.e.,* how nodes are connected to each other. In view of the involvement of graph, a naïve approach is that the value of $\mathbf{f}^i$ at every node v_i is the weighted average of $\mathbf{f}^i$ at neighbors of v_i, with the weights being proportional to the edge weights in adjacency matrix $\mathbf{A}$, which can be fitted as

$$\min_{\mathbf{F}} \left\| \mathbf{F} - \mathbf{D}'^{-1} \mathbf{A} \mathbf{F} \right\|_F^2 \tag{3.4}$$

Since there are some boundary-spanning nodes across clusters, and their neighbors naturally occur in different clusters, to exploit the formulation of (Equation 3.4) on interior-node clustering more effectively, it is crucial for the clustering indicator matrix $\mathbf{F}$ to have discriminative ability for such boundary-spanning nodes, *i.e.*, promoting row-wise sparsity to discriminate relevant boundary-spanning nodes, and thus achieving only characterizing interior nodes. Inspired by (He et al., 2016a), we introduce the $\ell_{2,1}$-norm penalty to make it and thus we have the following optimization problem:

$$\begin{aligned} &\min_{\mathbf{F}} \ \mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}'\mathbf{F}\right) + \alpha\left\|\mathbf{F} - \mathbf{D}'^{-1}\mathbf{A}\mathbf{F}\right\|_{2,1} \\ &\text{s.t.} \ \ \mathbf{F}^{\mathrm{T}}\mathbf{F} = \mathbf{I}_k \end{aligned} \tag{3.5}$$

where α is a parameter balancing two terms (*i.e.*, smoothness and sparsity). The sparsity-inducing property of $\ell_{2,1}$ norm will push the clustering indicator matrix $\mathbf{F}$ to be sparse in rows. More specifically, $\mathbf{f}^i$ will shrink to zero if the neighbors of node v_i belong to different clusters. In particular, the more nodes having neighbors belonging to different clusters, the larger $\left\|\mathbf{f}_i - \mathbf{D}'^{-1}\mathbf{A}\mathbf{f}_i\right\|_2^2$ tends to be, so the value of $\mathbf{f}^i$ gets penalized more harshly. We can thus obtain a better clustering indicator $\mathbf{F}$ for interior nodes.

As we discussed earlier, the graphs clustered into the same group tend to have more similar topological structure, in each iteration of our framework, we hope to further improve the interior-node clustering of each graph by incorporating the interior-node clustering results of its neighbors, *i.e.*, the graphs clustered into the same group by the multi-graph clustering stage

of the previous iteration. For two graphs in the same cluster, the closer they are, the more similar interior-node clustering they tend to have. Based on this assumption, for graph G_i, we consider only the graphs that are in the same cluster with G_i, and we aim to infer the weights of influence they should have on G_i.

Suppose we have a set of feature matrices $X = \{\mathbf{X}_1, \mathbf{X}_2, \cdots, \mathbf{X}_n\}$, where $\mathbf{X}_i$ can represent both the global and local structure of G_i, we aim to infer a weight matrix $\mathbf{W}$ by solving the following minimization problem:

$$\begin{aligned} &\min_{\mathbf{W}} \sum_i \left\| \mathbf{X}_i - \sum_j w_{ij} \mathbf{X}_j \right\|_F^2 \\ &\text{s.t.} \sum_j w_{ij} = 1 \end{aligned} \tag{3.6}$$

where w_{ij} denotes the weight of G_j for G_i, which will be used to control the extent that $\mathbf{F}_j$ will be used to influence $\mathbf{F}_i$ in the next iteration, and G_j can only be a graph from the cluster containing G_i. A larger w_{ij} implies a closer distance between G_i and G_j in the same cluster.

Now we can improve the interior-node clustering of each graph by adding a weighted influence from the neighbour graphs based on the multi-graph clustering. For a graph G_i, the interior-node clustering can be obtained by solving the following objective function extended from Equation 3.5:

$$\begin{aligned} &\min_{\mathbf{F}_i} \operatorname{Tr}\left(\mathbf{F}_i^{\mathrm{T}} \mathbf{L}_i \mathbf{F}_i\right) + \alpha \left\| \mathbf{F}_i - \mathbf{D}_i^{-1} \mathbf{A}_i \mathbf{F}_i \right\|_{2,1} + \beta \left\| \mathbf{F}_i - \sum_j w_{ij} \mathbf{F}_j \right\|^2 \\ &\text{s.t.} \quad \mathbf{F}_i^{\mathrm{T}} \mathbf{F}_i = \mathbf{I}_k \end{aligned} \tag{3.7}$$

where $\mathbf{A}_i$ is the weighted affinity matrix of G_i, $\mathbf{D}_i$ is the diagonal matrix, and $\mathbf{L}_i$ is the symmetric normalized Laplacian matrix.

With the two stages illustrated above, we can formulate the overall iterative process. We first obtain an initial multi-graph clustering indicator matrix $\mathbf{H}_0$ by Equation 3.2, where $\mathbf{S}$ is computed by Equation 3.1 with $\delta = 1$. Then we can infer the weight matrix $\mathbf{W}$ by solving Equation 3.6, which will be used for optimizing the interior-node clustering of each graph in Equation 3.7. With the resulted $\mathbf{F}_i$ for each graph G_i, a new similarity matrix can be created by Equation 3.1, which leads to another iteration of multi-graph clustering by Equation 3.2. The overall iterative algorithm with optimization solutions will be discussed in the following section.

3.3.2 Optimization

Since the minimization problem in Equation 3.2 is a typical spectral clustering problem, we can directly solve it by computing the first c generalized eigenvectors of the eigenproblem as illustrated in (Shi and Malik, 2000).

To solve the minimization problem (Equation 3.7), we propose an iterative algorithm based on the half-quadratic minimization (Nikolova and Ng, 2005) and the following lemma (He et al., 2012a).

Lemma 1. Let $\phi(.)$ be a function satisfying the conditions: $x \to \phi(x)$ is convex on R; $x \to \phi(\sqrt{x})$ is convex on R_+; $\phi(x) = \phi(-x), \forall x \in R$; $\phi(x)$ is C^1 on R; $\phi''(0^+) \geq 0$, $\lim_{x\to\infty} \phi(x)/x^2 = 0$. Then for a fixed $\|\mathbf{u}^i\|_2$, there exists a dual potential function $\varphi(.)$, such that

$$\phi(\|\mathbf{u}^i\|_2) = \inf_{p\in R}\{p\|\mathbf{u}^i\|_2^2 + \varphi(p)\} \tag{3.8}$$

where p is determined by the minimizer function $\varphi(.)$ with respect to $\phi(.)$.

Let $\mathbf{P}_i = \mathbf{F}_i - \mathbf{D}_i^{-1}\mathbf{A}_i\mathbf{F}_i$. According to the analysis for the $\ell_{2,1}$ norm in (He et al., 2012a), if we define $\phi(x) = \sqrt{x^2+\epsilon}$, we can replace $\|\mathbf{P}_i\|_{2,1}$ with $\sum_{j=1}^{n}\phi(\|\mathbf{p}_i^j\|_2)$. Thus, based on Lemma 1, we reformulate the objective function of Equation 3.7 as follows:

$$\begin{aligned} &\min_{\mathbf{F}_i} \ \mathrm{Tr}\left(\mathbf{F}_i^{\mathrm{T}}\mathbf{L}_i\mathbf{F}_i\right) + \alpha\mathrm{Tr}\left(\mathbf{P}_i^{\mathrm{T}}\mathbf{Q}\mathbf{P}_i\right) + \beta\left\|\mathbf{F}_i - \sum_j w_{ij}\mathbf{F}_j\right\|^2 \\ &\text{s.t.} \ \ \mathbf{F}_i^{\mathrm{T}}\mathbf{F}_i = \mathbf{I}_k \end{aligned} \tag{3.9}$$

where $\mathbf{Q} = Diag(\mathbf{q})$, and $\mathbf{q}$ is an auxiliary vector of the $\ell_{2,1}$ norm. The elements of $\mathbf{q}$ are computed by $q_j = \frac{1}{2\sqrt{\|\mathbf{p}_i^j\|_2^2+\epsilon}}$, where ϵ is a smoothing term and is usually set to be a small constant value (we set $\epsilon = 10^{-4}$ in this work).

The quadratic optimization problem with orthogonal constraint have been well studied, and can be solved by a lot of solvers (Absil et al., 2009; Wen and Yin, 2013). Here we employ the solver *Algorithm 2* in (Wen and Yin, 2013) to solve Equation 3.9, which is a more efficient optimization algorithm with publicly available code.

Another optimization problem we need to solve is Equation 3.6. In (Saul and Roweis, 2000), such a minimization problem with respect to vectors is solved as a constrained least squares problem for locally linear embedding. Since the Frobenius norm for matrices is a straightforward generalization of the l_2 norm for vectors, we can directly obtain the following equation based on the analysis in (Saul and Roweis, 2000).

$$\left\|\mathbf{X}_i - \sum_j w_{ij}\mathbf{X}_j\right\|_F^2 = \sum_{jr} w_{ij}w_{ir}\mathbf{C}_{jr} \tag{3.10}$$

where G_j and G_r denote two neighbors of G_i, *i.e.*, G_j and G_r are in the cluster containing G_i. $\mathbf{C}_{jr}$ is the local covariance matrix, which can be obtained by

$$\mathbf{C}_{jr} = \frac{1}{2}(M_j + M_r - m_{jr} - M_0) \tag{3.11}$$

where $m_{jr} = -s_{jr}$ denotes the squared distance between the jth and rth neighbors of G_i, thus can be obtained by Equation 3.1, $M_j = \sum_z m_{jz}$, $M_r = \sum_z m_{rz}$ and $M_0 = \sum_{jr} m_{jr}$. Then the optimal weights can be obtained by:

$$w_{ij} = \frac{\sum_r \mathbf{C}_{jr}^{-1}}{\sum_{lz} \mathbf{C}_{lz}^{-1}} \tag{3.12}$$

For details about the derivation of the above solution, readers can refer to (Saul and Roweis, 2000). Based on the above analysis, we summarize the overall optimization algorithm of MGCT in Algorithm 1.

Algorithm 1 MGCT

Input: $D = \{G_1, G_2, \cdots, G_n\}$, c, k
Output: Assignments to c clusters

Initialize $\mathbf{H}_0$ s.t. $\mathbf{H}_0^{\mathrm{T}}\mathbf{H}_0 = \mathbf{I}_c$;
while not converge **do**
  Compute $\mathbf{W}$ according to Equation 3.12;
  for $i = 1; i <= n; i++$ **do**
    Initialize $\mathbf{F}_{i0}$ s.t. $\mathbf{F}_{i0}^{\mathrm{T}}\mathbf{F}_{i0} = \mathbf{I}_k$,$t \leftarrow 0$;
    while not converge **do**
      Set $\mathbf{Q}_t \leftarrow Diag(\frac{1}{2\sqrt{\|\mathbf{p}_i^i\|_2^2+\epsilon}})$;
      Compute $\mathbf{F}_{it+1}$ by solving Equation 3.9;
      $t \leftarrow t + 1$;
    end while
  end for
  Update $\mathbf{H}$ by solving Equation 3.2;
  Cluster $\mathbf{H}$ by k-means;
end while

3.4 Experiments

In order to empirically evaluate the effectiveness of the proposed multi-graph clustering approach for brain network analysis, we test our model on real fMRI brain network data and compare with several state-of-the-art baselines.

3.4.1 Data Collection and Preprocessing

In this work, we use two real resting-state fMRI datasets as follows:

- *Human Immunodeficiency Virus Infection (HIV)*: This dataset is collected from Chicago Early HIV Infection Study in Northwestern University(Ragin et al., 2012a). The clinical cohort in this study includes 77 subjects, 56 of which are early HIV patients (positive) and the other 21 are seronegative controls (negative). The two groups did not differ in

the demographic characteristics including age, gender, racial composition and education level.

- *Attention Deficit Hyperactivity Disorder (ADHD)*: This dataset is collected from ADHD-200 global competition dataset[1], which contains the resting-state fMRI images of 768 subjects. Subjects are either ADHD patients or normal controls. In particular, the patient group in ADHD involves three stages of ADHD disease, which can be treated as three different groups, making the total number of groups be 4.

We use DPARSF toolbox [2] for fMRI data preprocessing. A time series of responds is extracted from each of the 116 anatomical volumes of interest (AVOI), which represents the 116 different brain regions. We perform the standard fMRI brain image processing steps, including functional images realignment, slice timing correction and normalization. Afterwards, spatial smoothing is performed on these images with an 8-mm Gaussian kernel for increasing signal-to-noise ratio, followed by the band-pass filtering (0.01-0.08 Hz) and the linear trend removing of the time series. We also apply linear regression to reduce spurious variance coming from hardware reasons or subject factors such as thermal motion of electrons. After all these preprocessing steps, we compute the brain activity correlations among different brain regions based on the obtained time series for each of them, and then we use the positive correlations to

---

[1]http://neurobureau.projects.nitrc.org/ADHD200/

[2]http://rfmri.org/DPARSF

form the links among the regions. Finally, we exclude the 26 cerebellar regions, and each brain is represented as a graph with 90 nodes, which correspond to the 90 cerebral regions.

3.4.2 Baselines and Metrics

We use four clustering methods as baselines.

- **k-means**: a classic clustering method (Berkhin, 2006). We convert the adjacency matrix of each subject graph into vectors and then apply the k-means algorithm to cluster all the subject graphs. For the implementation of the k-means algorithm, we adopt the Litekmeans (Cai, 2011), which has been proven to be a fast MATLAB implementation of the k-means algorithm.
- **Spectral Clustering (SC)** (Donath and Hoffman, 1973): a method for constructing graph partitions based on eigenvectors of the adjacency matrix of graph. In the experiment, we apply the normalized spectral clustering algorithm proposed in (Shi and Malik, 2000). We first construct the similarity matrix for the multiple graphs only based on their adjacency matrices and then use the similarity matrix as the input for normalized spectral clustering of the multiple graphs.
- **Clustering Coefficient (CC)**: the k-means clustering with clustering coefficient (Onnela et al., 2005) as the feature representation of each graph.
- **Two-layer Spectral Clustering(TSC)**: We adapt the typical spectral clustering into both of the two stages in our framework, where spectral clustering on the multi-graph is based on the spectral clustering on each graph. We call the model TSC.

- **MGCT**: our proposed multi-graph clustering method based on interior-node topology. To evaluate the discriminative ability of the sparsity-inducing term, *i.e.*, the $\ell_{2,1}$-norm penalty term in Equation 3.7, we employ MGCT with and without the sparsity-inducing term and denote them as **MGCT** and $\mathbf{MGCT}_{nonST}$ respectively.

We adopt the following two measures for the evaluation.

- *Accuracy*. Let c_i represent the clustering label result of a multi-graph clustering algorithm and y_i represent the corresponding ground truth label of the graph G_i. Then *Accuracy* is defined as: $Accuracy = \frac{\sum_{i=1}^{n} \delta(y_i, map(c_i))}{n}$, where δ is the Kronecker delta function, and $map(c_i)$ is the best mapping function that permutes clustering labels to match the ground truth labels using the KuhnMunkres algorithm (Kuhn, 1955). A larger *Accuracy* indicates better clustering performance.

- *Purity*. *Purity* is a measure used to evaluate the clustering method's ability to recover the groundtruth class labels, and it is defined as: $Purity = \frac{1}{n} \sum_{q=1}^{k} \max_{1 \leq j \leq l} n_q^j$, where n is the total number of samples, and n_q^j is the number of samples in cluster q that belongs to original class j. Therefore, the purity is a real number in $[0, 1]$. The larger the *Purity*, the better the clustering performance.

The main parameters in our framework include the weight parameters α, β, and δ as well as the number of interior-node clusters k. Note that in the rest part of this work, we use k specifically to denote the number of interior-node clusters in each graph although it might has been used for denoting other general variables in the equations above. For the convenience

TABLE III: Clustering *Accuracy*.

Methods	*Accuracy*	
	ADHD (k=6)	HIV (k=9)
k-means	52.0%	60.3%
Spectral Clustering	55.2%	60.9%
CC	56.8%	63.7%
TSC	57.6%	62.5%
MGCT$_{nonST}$	**59.3%**	**64.9%**
MGCT	**62.8%**	**68.1%**

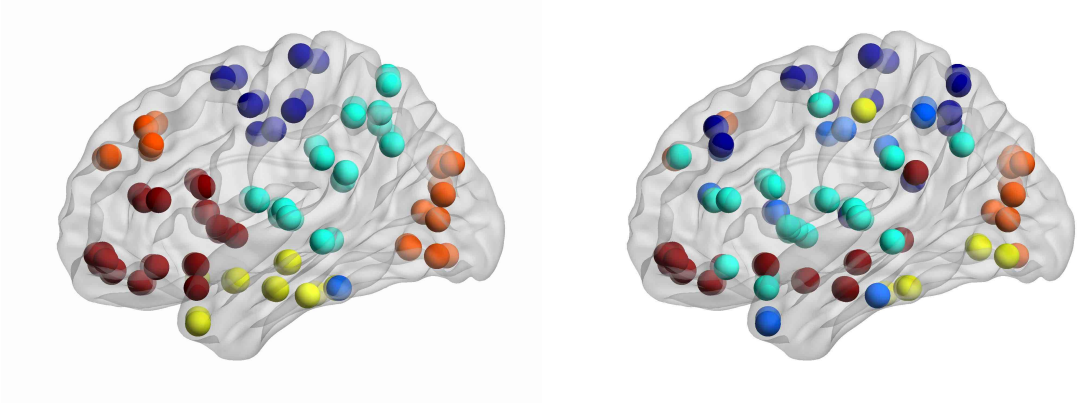
TABLE IV: Clustering *Purity*.

Methods	*Purity*	
	ADHD (k=6)	HIV (k=9)
k-means	0.55	0.63
Spectral Clustering	0.59	0.65
CC	0.57	0.66
TSC	0.57	0.64
MGCT$_{nonST}$	**0.62**	**0.69**
MGCT	**0.67**	**0.72**

of evaluation, we directly use the number of distinct labels in each dataset as the number of clusters in multi-graph clustering. Since there are four possible labels of the samples in ADHD datasets, we set the number of clusters to be 4. For HIV dataset, we have two possible labels (positive, negative), so we set the cluster number to be 2. We apply the grid search to find the optimal values for α, β and δ. We do grid search for α in $\{10^{-2}, 10^{-1}, \cdots 10^{2}\}$, β in $\{10^{-4}, 10^{-3}, \cdots 10^{4}\}$, and δ in $\{0.1, 0.2, \cdots 0.9\}$. The optimal k is selected by the grid search from $\{2, 3, \cdots, 12\}$. For fair comparisons of all the methods, we employ Litekmeans (Cai, 2011) for all the k-means clustering step if it is needed in the implementation of the six methods listed above. We repeat clustering for 20 times with random initialization as k-means depends on initialization. For the evaluation, we repeat running the program of each methods for 50 times and report the average accuracy and purity as the results.

3.4.3 Performance Evaluations

As shown in Table III and Table IV, our MGCT method performs the best on the two datasets in terms of both *accuracy* and *purity*. Among the six clustering methods, the first two methods (*i.e.*, k-means, Spectral Clustering) directly use the original matrix of each graph in



(a) a typical normal control (b) a stage-2 ADHD patient

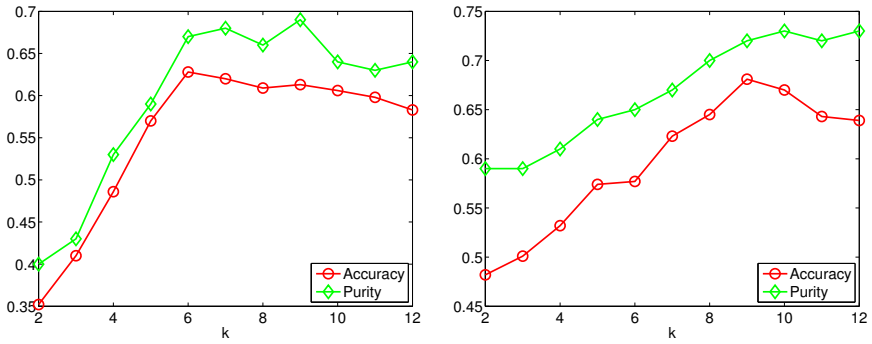
Figure 7: Comparison of two brain networks with interior-node topology captured by MGCT from two subject graphs in ADHD dataset

the data set for calculating the distance or similarity between each pair of the graphs, which is utilized for the final multi-graph clustering. From Table III and Table IV, we can see that the clustering accuracy and purity of these two methods are quite low. This is probably because that they do not consider the interior-node topology of these graphs when doing clustering. The CC achieves a slightly better result compared to k-means and Spectral Clustering. This is mainly due to the fact that CC does consider some local structure information while calculating the clustering coefficient. However, since it only assigns a single predefined measure to each node in the graph, which represents each brain region in the brain networks, the subtle topological structure of each brain network might not be fully characterized.

Comparatively, the last three methods (*i.e.*, TSC, MGCT, MGCT$_{nonST}$) all utilize the topological structure information but at different level. The TSC method first performs spectral

clustering on each graph, and the resulted matrix containing the clustering indicator vectors are used in the multi-graph spectral clustering. This process does include the topological structure, but it only has the one-way and one-time influence on the multi-graph clustering task. The result of multi-graph clustering does not have influence on the interior-node clustering. Different from TSC, the two methods we proposed namely the MGCT and MGCT$_{nonST}$ perform the task in an iterative way, and achieves the mutual reinforcement by leveraging the topology structure into multi-graph clustering and inferring a better topology structure for each graph from the multi-graph clustering result alternatively. According to Table III and Table IV, we can also see that the proposed MGCT method outperforms the MGCT$_{nonST}$ in both accuracy and purity. This indicates the importance of the $\ell_{2,1}$ norm we add in Equation 3.7, which has the sparsity-inducing property.

In order to evaluate the effectiveness of MGCT for interior-node topology extraction of brain networks, we investigate the resulted brain networks with interior-node clusters detected by MGCT and show the results of two brain networks in Figure 7. We can find from the figure that the interior nodes of the normal brain network have been well grouped into several clusters, while the cluster boundaries in the patient's brain network are very blurred and the nodes widely spread out. Usually, the correlated regions of human brain will work together towards a task, and tend to present an approximately synchronized trend in their time series. Thus, the nodes representing these correlated regions would become more possible to be grouped into the same cluster. Therefore, the fuzzy cluster boundaries of the patient's interior nodes indicate that the collaboration activity of different regions might not be very organized. These



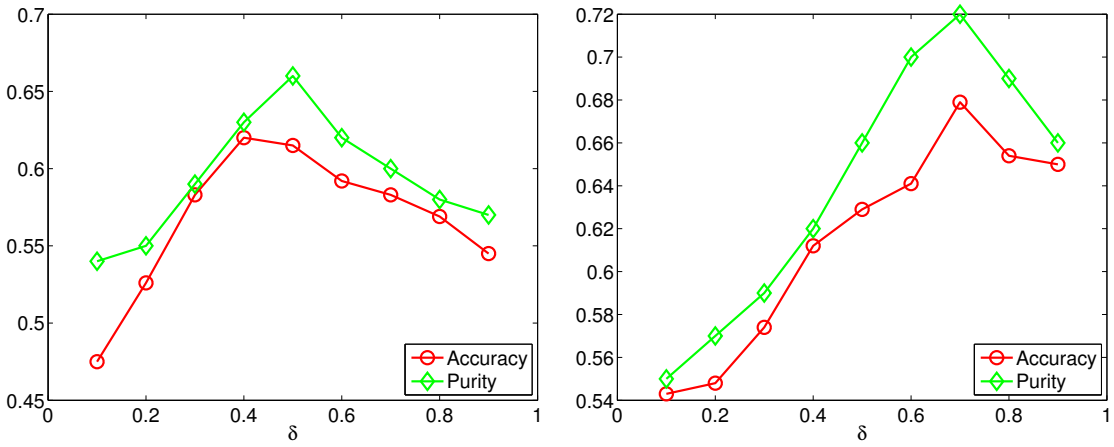
(a) ADHD (b) HIV

Figure 8: Accuracy and purity with different k

observations imply that our proposed framework can be further used for distinguishing subjects with neurological disorders from healthy controls.

3.4.4 Parameter Sensitivity

In this section, we explore the sensitivity and effects of the four main parameters in our proposed method, including α, β, δ and k. We first evaluate the clustering performance of MGCT with different k values, ranging from 2 to 12. Figure 8 shows the clustering performance of MGCT in accuracy and purity with different k on both ADHD and HIV datasets. As we can see from the figure, the multi-graph clustering performance is very sensitive to the value of k, especially when the value for k keeps very small. For example, as shown in 8(a), the accuracy increases dramatically when the value of k goes from 2 to 6 before it reaches the peak value at 6. The main reason for such high sensitivity is that when k is set to be a small number, the interior-node clusters identified from each brain network tend to have large sizes, which could not capture the interior-node topological structure very well, resulting in a less discriminative



(a) ADHD (b) HIV

Figure 9: Accuracy and purity with different δ

measure for distinguishing subjects in different neurological states. A similar changing trend is shown for the purity, while noticeably the peak purity value shows up when $k = 9$ instead of $k = 6$. This can be traced back to the definition of *purity*. Since it counts the number of nodes in the dominated class for each cluster instead of counting the number of nodes only when they match the correct groundtruth labels. Thus, when the number of clusters increases, each cluster becomes easier to be dominated by one class, leading to a higher purity.

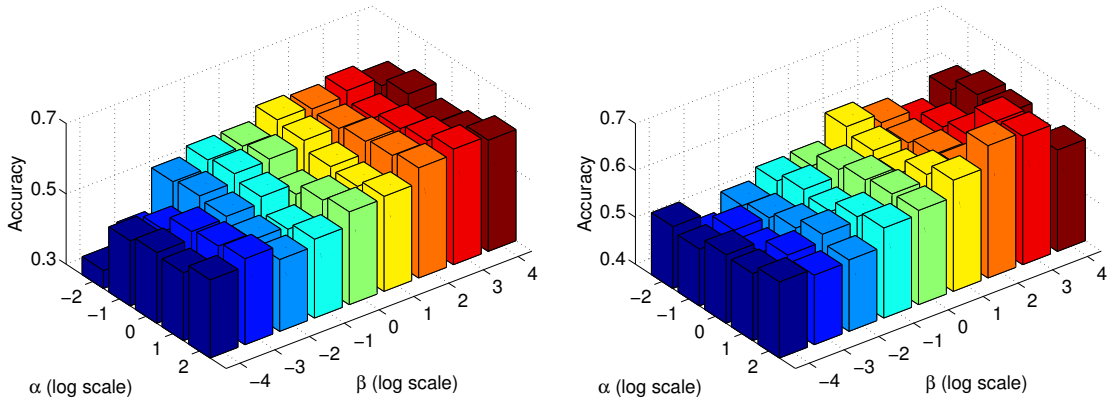
Now, we analyze the sensitivity of MGCT to δ, which balances the weights from the original affinity matrix and the interior-node clustering indicator matrix when creating the similarity matrix among multiple graphs. As shown in Figure 9, MGCT achieves different level of accuracy and purity when the value of δ varies. For ADHD, the highest accuray is achieved when $\delta = 0.4$, while for HIV, it achieves the highest accuracy when $\delta = 0.7$, and similar situations for the

purity. These results indicate that both the global structure and the interior-node topological structure are important for the multi-graph clustering analysis, and their weights need to be determined for specific practical situations. Next, we evaluate the sensitivity of MGCT to α and β. We set k to be 6 and run the MGCT method with different values for α and β on ADHD and HIV data. The clustering accuracy of MGCT is plotted versus the values for α and β in Figure 10. As shown in the figure, MGCT achieves the best performance when $\alpha = 10^2, \beta = 10^3$ on ADHD dataset, and $\alpha = 10^2, \beta = 10^2$ on HIV dataset. Parameter α controls the sparsity while parameter β controls the influence of iterative multi-graph clustering results on interior-node clustering. If the value for α is very small, then it will not really enforce the sparsity. Similarly, if the value for β is quite small, the iterative process would barely have influence on interior-node clustering optimizing. In these cases, the performance will decline. However, when the values for them are too large, they would enforce too much sparsity or influence, which might make the performance drop as well. Therefore, an optimal combination of the two parameters is crucial for improving the performance of MGCT.

3.5 Related Work

Our work relates to several bodies of studies, including the multi-graph clustering, node clustering in graphs, and brain network analysis.

In the context of multi-graph clustering, there are a few of strategies that have been proposed and widely used (Aggarwal and Wang, 2010), for example the structural summary method discussed in (Aggarwal et al., 2007), and the hierarchical algorithm with graph structure proposed in (Lian et al., 2004). However, these methods only focus on finding a summary from the global



(a) ADHD (b) HIV

Figure 10: Accuracy with different α, β

structure of the graphs without looking into the topological structure, thus would lose very important local structural information, leading to a less effective clustering of multiple graphs.

For node clustering in graphs, there has also been a vast literature of works. One classic category of these methods are the spectral clustering algorithms (Von Luxburg, 2007), which use the eigenvalues of the Laplacian matrix to perform dimension reduction and then cluster the data in fewer dimensions. Recently, new methods of node clustering have been proposed for various applications, such as the works for social network analysis (Zhang and Yu, 2015a; Zhang and Yu, 2015b), which utilize the heterogeneous information in aligned networks for node clustering. Although these work use information from multiple graphs, they focus on the mutual relationship of graphs at the node level instead of the graph-graph neighbourhood relationship as we consider.

Brain network analysis has become a hot research topic of medical data mining these years. A major task in brain network analysis is to identify the difference of a healthy subject and a neurological demented subject in brain network structure. In the past decade, quite a few of works have been done to solve this problem. In (Kong et al., 2013), a discriminative subgraph mining method is proposed for classifying brain networks. In (Kuo et al., 2015), they find a unified cut and a contrast cut of multiple graphs for studying brain networks of multiple subjects. This work is the most related one of ours. However, they study the brain networks when the labels of subjects (healthy or demented) are given, while we cluster the unlabeled subjects into groups with their brain network features.

CHAPTER 4

MULTI-VIEW CLUSTERING WITH GRAPH EMBEDDING

(This chapter was previously published as "Multi-view Clustering with Graph Embedding for Connectome Analysis", in CIKM '17 (Ma et al., 2017). DOI: https://doi.org/10.1145/3132847.3132909.)

4.1 Introduction

Advances in capabilities for data acquisition have given rise to an explosion of new information in the form of graph representations. These data are inherently represented as a set of nodes and links, instead of feature vectors as in traditional data. Brain networks, for example, are comprised of anatomic regions as nodes, and functional/structural connectivities between the brain regions as links. Linkage structures often come from different sources, called as multi-view data. For instance, fMRI (functional magnetic resonance imaging) and DTI (diffusion tensor imaging) are two major neuroimaging approaches widely used in neuroscience research and in clinical applications (Cao et al., 2015a; Zhang et al., 2016; Ma et al., 2016). Connections in brain networks derived from fMRI brain images encode correlations in functional activity among brain regions, whereas DTI networks provide information concerning structural connections (*i.e.* white matter fiber paths) between different brain regions. The different networks afford two different views of the brain connectivity.

Multi-view clustering has received considerable attention for unlabeled data with multiple views from diverse domains. While there have been advances in multi-view clustering, most approaches are based on vector representation of features in each view and combining vectors from different views for the clustering task (Liu et al., 2013; Yin et al., 2016). However, the complex structures and the lack of vector representations within graph data, pose serious challenges for this kind of vector-based approach. It is desirable to find a way that can better capture and exploit graph structural information for multi-view clustering of graph instances. To address this problem, this work explores an approach involving multi-view clustering of graph instances based on graph embedding and its application to connectome analysis in multi-view brain networks on HIV disease. The goal of graph embedding is to find low-dimensional representations of graphs that can preserve the inherent structure and properties (Yan et al., 2007; Mousazadeh and Cohen, 2015). While graph embedding technology has been broadly used for graph mining, to the best of our knowledge, this approach has not been used for multi-view clustering of graph instances. There are two main challenges that must be addressed for the problem of multi-view clustering with graph embedding:

- How to learn the graph embedding for each graph instance with multiple views, such that the graph embedding can encode the multi-view structure information of the graph? Specifically, the embeddings of the similar nodes within the graph instance should be close.
- How to leverage the multi-view graph embedding results to facilitate the multi-view clustering task on graph instances? The graph embedding mainly captures the local structure

of graphs, while the similarity between the graph instances holds their global structure information. For multi-view clustering on graph data, it is critical to appropriately fuse these two kinds of graph structure information .

To address the above challenges, we propose the MCGE (multi-view clustering with graph embedding) framework. Our contributions can be summarized as:

- We model the multi-view graph data with tensors, and apply tensor factorization technique to learn the multi-view embedding of graphs. In this manner, the graph embedding can capture the key local structure of the graph in all the views, while also encoding the latent correlation between different views. We employ graph kernel to measure the similarity between graph instances in each view, construct a multi-view kernel tensor based on kernel matrices, and obtain the common latent factors that encode the global structure information.
- We propose to jointly perform the multi-view graph embedding stage and the multi-view clustering stage in an iterative manner. Considering the fact that the graphs clustered into the same group tend to have similar local structure, for each graph, we use the multi-view embeddings of the neighbour graphs clustered in the same group to refine its multi-view embedding. Then the updated multi-view embedding of the graphs will be used for the multi-view clustering stage in the next iteration. Following this iterative two-stage process, the multi-view graph embedding and multi-view clustering will be improved until we obtain an optimal clustering results.

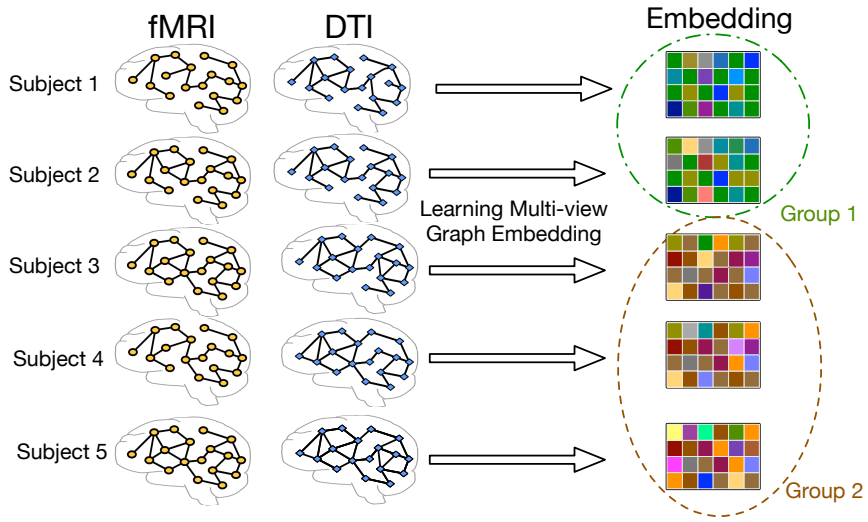


Figure 11: An example of the MCGE problem

- We apply the proposed MCGE framework for unsupervised multi-view connectome analysis on HIV and Bipolar. Specifically, we study the connectome of fMRI and DTI brain networks and aim to cluster the subjects with similar neurological status into the same group as shown in Figure 11. Experimental results on the HIV and Bipolar datasets show the effectiveness of MCGE for multi-view clustering in connectome analysis.

4.2 Preliminaries

In this section, we first introduce some notations and terminologies that we will use throughout the work. Then we formulate the problem of interest formally.

Notations. Vectors are denoted by boldface lowercase letters, matrices are denoted by boldface capital letters, and tensors are denoted by calligraphic letters. An element of a vector $\mathbf{x}$, a matrix $\mathbf{X}$, or a tensor $\mathcal{X}$ is denoted by x_i, x_{ij}, x_{ijk}, etc., depending on the number of indices

TABLE V: List of basic symbols.

Symbol	Definition and description
x	each lowercase letter represents a scale
$\mathbf{x}$	each boldface lowercase letter represents a vector
$\mathbf{X}$	each boldface uppercase letter represents a matrix
$\mathcal{X}$	each calligraphic letter represents a tensor
$\langle \cdot, \cdot \rangle$	denotes inner product
$\circ$	denotes tensor product (outer product)
$\otimes$	denotes Kronecker product
$\odot$	denotes Khatri-Rao product

(also known as modes). For a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, its i-th row and j-th column are denoted by $\mathbf{x}^i$ and $\mathbf{x}_j$, respectively. The Frobenius norm of $\mathbf{X}$ is defined as $\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^{n} \|\mathbf{x}^i\|_2^2}$. For any vector $\mathbf{x} \in \mathbb{R}^n$, $Diag(\mathbf{x}) \in \mathbb{R}^{n \times n}$ is the diagonal matrix whose diagonal elements are x_i. $\mathbf{I}_n$ denotes an identity matrix with size n. We denote an undirected graph as $G = (V, E)$, where V is the set of nodes and $E \subset V \times V$ is the set of edges. An overview of the basic symbols used in this work can be found in Table V.

Definition 1 (Tensor)**.** An nth-order tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times I_2 \times \cdots \times I_n}$ is an element of the outer product of n vector spaces, each of which has its own coordinate system.

Definition 2 (Outer product)**.** The outer product of vectors $\mathbf{x}^{(k)} \in \mathbb{R}^{I_k}$ for $k = 1, 2, \cdots, n$ is an n-th order tensor and defined elementwise by $\left(\mathbf{x}^{(1)} \circ \mathbf{x}^{(2)} \circ \cdots \circ \mathbf{x}^{(n)}\right)_{i_1, i_2, \cdots, i_n} = x_{i_1}^{(1)} x_{i_2}^{(2)} \cdots x_{i_n}^{(n)} = \prod_{k=1}^{n} x_{i_k}^{(k)}$ for all values of the indices.

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 a matrix in the dimension of $IK \times JL$:

$$\mathbf{A} \otimes \mathbf{B} = \begin{pmatrix} a_{11}\mathbf{B} & a_{12}\mathbf{B} & \cdots & a_{1J}\mathbf{B} \\ a_{21}\mathbf{B} & a_{22}\mathbf{B} & \cdots & a_{2J}\mathbf{B} \\ \vdots & \vdots & \vdots & \vdots \\ a_{I1}\mathbf{B} & a_{I2}\mathbf{B} & \cdots & a_{IJ}\mathbf{B} \end{pmatrix} \tag{4.1}$$

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 a matrix in dimension of $IJ \times K$:

$$\mathbf{A} \odot \mathbf{B} = (a_1 \otimes b_1, a_2 \otimes b_2, \cdots, a_K \otimes b_K) \tag{4.2}$$

where $a_1, a_2, \cdots, a_K$ are the columns of $\mathbf{A}$ and $b_1, b_2, \cdots, b_K$ are the columns of $\mathbf{B}$.

Definition 5 (Mode-k Matricization)**.** The mode-k matricization of a tensor $\mathcal{X} \in \mathbb{R}^{I_1\times I_2\times\cdots\times I_n}$, denoted by $\mathbf{X}_{(k)} \in \mathbb{R}^{I_k\times J}$, where $J = \Pi_{q=1,q\neq k}^{n} I_q$. Each tensor element with indices $(i_1, i_2, \cdots, i_n)$ maps to a matrix element (i_k, j), such that

$$\begin{aligned} j &= 1 + \sum_{p=1,p\neq k}^{m} (i_p - 1)J_p, \ with \\ J_p &= \begin{cases} 1, & \text{if } p = 1 \text{ or } (p = 2 \text{ and } k = 1) \\ \Pi_{q=1,q\neq k}^{p-1} I_q, & \text{otherwise.} \end{cases} \end{aligned} \tag{4.3}$$

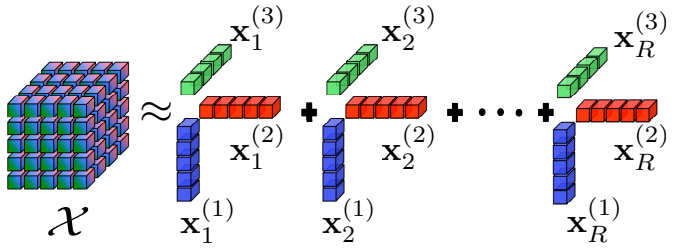


Figure 12: The CP Factorization for a third-order tensor $\mathcal{X}$

Definition 6 (CP Factorization)**.** Given a tensor $\mathcal{X} \in \mathbb{R}^{I_1 \times \cdots \times I_n}$, its CANDECOMP / PARAFAC (CP) factorization is

$$\mathcal{X} = [\![\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}]\!] \equiv \sum_{r=1}^{R} \mathbf{x}_r^{(1)} \circ \cdots \circ \mathbf{x}_r^{(n)}, \tag{4.4}$$

where for $k = 1, 2, \cdots, n$, $\mathbf{X}^{(k)} = [\mathbf{x}_1^{(k)}, \cdots, \mathbf{x}_R^{(k)}]$ are factor matrices of size $I_k \times R$, R is the number of factors, and $[\![\cdot]\!]$ is used for shorthand. Figure 12 shows the form of the CP Factorization for a third-order tensor example.

To obtain the CP factorization $[\![\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}]\!]$, the objective is to minimize the following estimation error:

$$\mathcal{L} = \min_{\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}} \|\mathcal{X} - [\![\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}]\!]\|_F^2 \tag{4.5}$$

However, $\mathcal{L}$ is not jointly convex w.r.t. $\mathbf{X}^{(1)}, \cdots, \mathbf{X}^{(n)}$. A widely used optimization technique is the Alternating Least Squares (ALS) algorithm, which alternatively minimize $\mathcal{L}$ for each variable while fixing the other, that is,

$$\mathbf{X}^{(k)} \leftarrow \arg\min_{\mathbf{X}^{(k)}} \|\mathbf{X}_{(k)} - \mathbf{X}^{(k)}(\odot_{i\neq k}^{n}\mathbf{X}^{(i)})^{\mathrm{T}}\|_F^2 \tag{4.6}$$

where $\odot_{i\neq k}^{n}\mathbf{X}^{(i)} = \mathbf{X}^{(n)} \odot \cdots \mathbf{X}^{(k+1)} \odot \mathbf{X}^{(k-1)} \cdots \odot \mathbf{X}^{(1)}$.

Problem Definition We study the problem of multi-view clustering of graph instances via multi-view graph embedding. Assume we are given a set of instances $D = \{G_1, G_2, \cdots, G_n\}$ with v views, where each instance is represented with a graph with m nodes in each view. For the j-th view, we have a set of graphs with the affinity matrices $D^{(j)} = \{\mathbf{G}_1^{(j)}, \mathbf{G}_2^{(j)}, \cdots, \mathbf{G}_n^{(j)}\}$. The goal of multi-view clustering on D is to cluster the graphs in D into k subsets. Figure 11 shows a simple two-view example of the MCGE problem intuitively. Given the fMRI and DTI brain networks of five subjects, MCGE aims to learn multi-view graph embedding for each of them, and cluster these subjects into different groups based on the obtained multi-view graph embedding.

4.3 MCGE Framework

In this section, we first present the proposed MCGE framework consisting of two stages: multi-view graph embedding and multi-view clustering via graph embedding. Then we describe the optimization scheme of our framework.

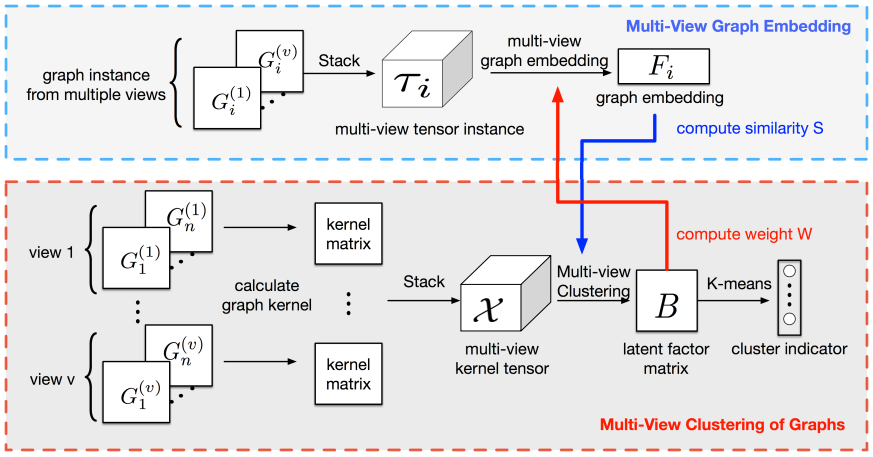


Figure 13: The framework of the proposed model MCGE.

4.3.1 Multi-view Graph Embedding

Graph embedding is an important tool in topological graph theory, which has been widely used in data analysis (Belkin and Niyogi, 2001; Fu and Ma, 2012; Yan et al., 2007). In the unsupervised situation, conventional methods for multi-view graph embedding either glued the graph affinity matrices from all the views together into a big graph (Cilla Ugarte, 2012; Gao et al., 2013), or collaboratively explored the consensus embedding from different views (individual affinity matrices) (Xia et al., 2010; Xie et al., 2011; Zhang et al., 2015). However these methods can only capture the linear relationships in multi-view graph data. In order to achieve better embedding, here we develop a multilinear embedding approach via tensorization as follows.

To model the multiple views for each graph instance G_i, we build a tensor $\mathcal{T}_i$ by stacking the graph affinity matrices from all the v views of the graph. Assume that the dimension of the row vectors in the graph embedding is c, and let $\mathbf{F}_i \in \mathbb{R}^{m \times c}$ be the graph embedding of G_i,

i.e., the j-th row vector of $\mathbf{F}_i$ represent the embedding of node j on graph instance G_i. Then we can formulate the multi-view graph embedding as the following optimization problem based on CP factorization:

$$\begin{aligned} \min_{\mathbf{F_i},\mathbf{H_i}} \quad & \|\mathcal{T}_i - [\![\mathbf{F}_i, \mathbf{F}_i, \mathbf{H}_i]\!]\|_F^2 \\ \text{s.t.} \quad & {\mathbf{F}_i}^{\mathrm{T}}\mathbf{F}_i = \mathbf{I}_c \end{aligned} \tag{4.7}$$

where $\mathbf{F}_i \in \mathbf{R}^{m\times c}$ and $\mathbf{H}_i \in \mathbf{R}^{v\times c}$ are the latent factor matrices.

Besides, as we discussed earlier, the graphs clustered into the same group tend to have more similar local structure. That is, for two graphs in the same cluster, the closer they are, the more similar local structure they tend to have. Based on this assumption, we incorporate such global cluster information to further improve the multi-view graph embedding result in Equation 4.7. Assuming we can obtain a weight matrix $\mathbf{W}$, where w_{ij} denotes the weight of G_j for G_i and a larger w_{ij} implies a closer distance between G_i and G_j in the same cluster. By incorporating the weighted influence from the neighbor graphs into Equation 4.7, we have the following objective function:

$$\begin{aligned} \min_{\mathbf{F_i},\mathbf{H_i}} \quad & \|\mathcal{T}_i - [\![\mathbf{F}_i, \mathbf{F}_i, \mathbf{H}_i]\!]\|_F^2 + \beta\left\|\mathbf{F}_i - \sum_j w_{ij}\mathbf{F}_j\right\|_F^2 \\ \text{s.t.} \quad & {\mathbf{F}_i}^{\mathrm{T}}\mathbf{F}_i = \mathbf{I}_c \end{aligned} \tag{4.8}$$

where β is a parameter balancing the two parts.

In the following section, we will show how to incorporate the graph embeddings into the multi-view clustering framework and how to obtain the weight matrix $\mathbf{W}$ from the clustering results.

4.3.2 Multi-view Clustering via Graph Embedding

Since graph embedding usually encodes local structure of graphs, and the original affinity matrix holds the global structure, we propose to consider both of these two kinds of structure information for the multi-view clustering task. Specifically, we employ the graph kernel to measure the similarity of the global structure between different graphs. Graph kernel is a pervasive method for comparing graphs (Vishwanathan et al., 2010). Here we employ the random walk graph kernel (Vishwanathan et al., 2010), which is one of the most widely used graph kernels, to measure the similarity between the affinity matrices of different graphs in each view. Since we have n graphs in v views, we will get v kernel matrices, each with dimension of $n \times n$. In order to integrate the multiple views, we propose to stack the v kernel matrices together, which form a tensor $\mathcal{X} \in \mathbf{R}^{n \times n \times v}$. Then we apply CP factorization on the tensor $\mathcal{X}$ to get the common factor matrices across all the views. Suppose the number of factors is k, $\mathcal{X}$ can be factorized as:

$$\mathcal{X} = [\![\mathbf{B}, \mathbf{B}, \mathbf{A}]\!] \tag{4.9}$$

where $\mathbf{B} \in \mathbf{R}^{n \times k}$ and $\mathbf{A} \in \mathbf{R}^{v \times k}$ are the latent factor matrices. Notably, $\mathbf{B}$ can be interpreted as the common latent factor across all the views, which can be used for clustering the graphs.

Now let us consider how to incorporate the results of multi-view graph embedding into the multi-view clustering stage. As we discussed above, the multi-view graph embedding implies the local structure of the graph, and graphs with similar local structure tend to be close to each other in the original multi-view feature space.

Suppose we have obtained a set of graph embeddings $F = \{\mathbf{F}_1, \mathbf{F}_2, \cdots, \mathbf{F}_n\}$, where $\mathbf{F}_i \in \mathbb{R}^{m \times c}$ is the multi-view graph embedding for G_i, we can build a similarity matrix $\mathbf{S} \in \mathbb{R}^{n \times n}$, where s_{ij} denotes the similarity between two examples G_i and G_j in terms of graph embedding, and we define it as:

$$s_{ij} = 1 - \|\mathbf{F}_i - \mathbf{F}_j\|_F^2 \tag{4.10}$$

Then we can formulate the following objective function on the basis of the spectral analysis (Von Luxburg, 2007):

$$\begin{aligned} \min_{\mathbf{B}} \quad & = \sum_{i,j=1}^{n} s_{ij} \left\| \frac{\mathbf{b}_i}{\sqrt{d_{ii}}} - \frac{\mathbf{b}_j}{\sqrt{d_{jj}}} \right\|_2^2 = \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}} \mathbf{L} \mathbf{B}\right) \\ \text{s.t.} \quad & \mathbf{B}^{\mathrm{T}} \mathbf{B} = \mathbf{I}_k \end{aligned} \tag{4.11}$$

where $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{S})\mathbf{D}^{-\frac{1}{2}}$ is the symmetric normalized Laplacian matrix, and $\mathbf{D}$ is a diagonal matrix with $d_{ii} = \sum_{j=1}^{n} s_{ij}$.

By combining the above tensor CP factorization strategy with Equation 4.11, we can formulate the multi-view clustering task as the following optimization problem:

$$\begin{aligned} \min_{\mathbf{B},\mathbf{A}} \ & \|\mathcal{X} - [\![\mathbf{B}, \mathbf{B}, \mathbf{A}]\!]\|_F^2 + \alpha \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}\right) \\ \text{s.t.} \quad & \mathbf{B}^{\mathrm{T}}\mathbf{B} = \mathbf{I}_k \end{aligned} \tag{4.12}$$

where α is a parameter balancing two parts.

After we obtain matrix $\mathbf{B}$, we can apply k-means clustering on the row vectors of $\mathbf{B}$ and then we know which graphs are clustered into the same group and which ones are not. This result will help determine the weight matrix $\mathbf{W}$ for the multi-view graph embedding stage. Specifically, for graph G_i we consider the graphs from the same cluster with G_i, and we aim to infer the weights of influence they should have on G_i. Suppose we use $\mathbf{X}_i$ to represent both the global and local structure of G_i,then this problem can be formulated as the following minimization problem based on LLE method (Roweis and Saul, 2000):

$$\begin{aligned} \min_{\mathbf{W}} \ & \sum_i \left\| \mathbf{X}_i - \sum_j w_{ij}\mathbf{X}_j \right\|_F^2 \\ \text{s.t.} \quad & \sum_j w_{ij} = 1 \end{aligned} \tag{4.13}$$

where w_{ij} denotes the weight of G_j for G_i, and $w_{ij} = 0$ if G_j and G_i are not in the same cluster. Note that there is no need for an explicit definition of $\mathbf{X}_i$ here, as it will be implicitly represented with both the affinity matrices and the multi-view graph embedding results, which will be used for the optimization of $\mathbf{W}$. The details will be illustrated in Section 4.4.

4.3.3 The Overall Framework: MCGE

With the two stages discussed above, we can formulate the overall iterative process for the MCGE framework. As the multi-view graph embedding and multi-view clustering depend on each other, we propose to jointly perform these two stages. In each iteration, we first perform the multi-view graph embedding on each graph, and then utilize the obtained graph embedding in the multi-view clustering stage. Then the resulted graph cluster information will be used for refining the multi-view graph embedding in the next iteration. Following this alternate two-stage process, the multi-view graph embedding and multi-view clustering will be improved by each other until convergence.

An overview of our framework is shown in Figure 13. The upper part demonstrates the multi-view graph embedding stage in MCGE, and the lower part shows the multi-view clustering stage, while the blue arrow and red arrow indicate the interaction of the two stages. Overall, given a set of graph instances $D = \{G_1, G_2, \cdots, G_n\}$ with v views, we aim to obtain a multi-view graph embedding for each of these graph instances, and then use the multi-view graph embedding as key features for the clustering of graph instances.

As shown in Figure 13, in the multi-view graph embedding stage, for each graph instance G_i, we stack its affinity matrices from all the v views together to form a multi-view tensor instance $\mathcal{T}_i$. Then we apply tensor factorization strategy in Equation 4.8 to learn the multi-view embedding, which partially depends on the embedding of the other graphs from the same cluster that is determined by the multi-view clustering stage. Meanwhile, in the multi-view clustering stage, we first measure the similarity between each pair of the graphs by calculating

the graph kernel from each view, and then we stack the kernel matrices from all the views, resulting in a multi-view kernel tensor $\mathcal{X}$. By utilizing the CP factorization on $\mathcal{X}$, we can get the common factor $\mathbf{B}$ across all the views. Considering the importance of graph embedding in capturing graph structure, we compute the similarity between graphs based on the multi-view graph embedding results and incorporate it into the CP factorization scheme with a spectral analysis term, as shown in Equation 4.12. The latent factor $\mathbf{B}$ obtained from this step will indicate which graphs are closer to each other, thus can be further used for computing the weight matrix $\mathbf{W}$, which will be used for updating the multi-view graph embedding in the next iteration. Vice versa, the new multi-view graph embedding will be used for updating the similarity $\mathbf{S}$, thus improving the multi-view clustering stage.

4.4 Optimization

Since the objective function in Equation 4.12 is not convex with respect to $\mathbf{A}$ and $\mathbf{B}$ jointly, and Equation 4.8 is not convex with respect to $\mathbf{F}_i$, there is no closed-form solution for such problem. We employ an Alternating Direction Method of Multipliers (ADMM) scheme (Boyd et al., 2011; Wang et al., 2015) to solve these problems, which alternately update one variable while fixing others until convergence.

We first solve the optimization problem in Equation 4.12. The variables to be estimated include $\mathbf{B}$ and $\mathbf{A}$.

Update factor matrix B. We first update $\mathbf{B}$ while fixing $\mathbf{A}$. Due to the fourth-order term, the objective function in Equation 4.12 is not convex with respect to $\mathbf{B}$, thus being difficult to

optimize. We employ the variable substitution technique to solve this problem. By substituting the second $\mathbf{B}$ with $\mathbf{P}$ in Equation 4.12, we obtain the equivalent form of Equation 4.12:

$$\begin{aligned} &\min_{\mathbf{B}} \|\mathcal{X} - [\![\mathbf{B}, \mathbf{P}, \mathbf{A}]\!]\|_F^2 + \alpha \mathrm{Tr}(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}) \\ &\text{s.t.} \quad \mathbf{P} = \mathbf{B},\ \mathbf{B}^{\mathrm{T}}\mathbf{B} = \mathbf{I}_k \end{aligned} \tag{4.14}$$

where $\mathbf{P}$ is auxiliary variable. The augmented Lagrangian function for Equation 4.14 is:

$$\begin{aligned} \mathscr{L}(\mathbf{B}, \mathbf{P}) = &\|\mathcal{X} - [\![\mathbf{B}, \mathbf{P}, \mathbf{A}]\!]\|_F^2 + \mathrm{Tr}\left(\mathbf{U}^{\mathrm{T}}(\mathbf{P} - \mathbf{B})\right) \\ &- \frac{\mu}{2}\|\mathbf{P} - \mathbf{B}\|_F^2 + \alpha \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}\right) \end{aligned} \tag{4.15}$$

where $\mathbf{U} \in \mathbf{R}^{n \times k}$ are Lagrange multipliers, and μ is the penalty parameter. Then the objective function with respect to $\mathbf{B}$ can be derived as:

$$\begin{aligned} &\min_{\mathbf{B}} \left\|\mathbf{B}\mathbf{Q}^{\mathrm{T}} - \mathbf{X}_{(1)}\right\|_F^2 + \frac{\mu}{2}\left\|\mathbf{B} - \mathbf{P} - \frac{1}{\mu}\mathbf{U}\right\|_F^2 + \alpha \mathrm{Tr}\left(\mathbf{B}^{\mathrm{T}}\mathbf{L}\mathbf{B}\right) \\ &\text{s.t.} \quad \mathbf{B}^{\mathrm{T}}\mathbf{B} = \mathbf{I}_k \end{aligned} \tag{4.16}$$

where $\mathbf{Q} = \mathbf{P} \odot \mathbf{A} \in \mathbf{R}^{(n*v) \times k}$ and $\mathbf{X}_{(1)} \in \mathbf{R}^{n \times (n*v)}$ is the mode-1 matricization of $\mathcal{X}$.

As such an optimization problem with orthogonal constraint has been well studied, and can be solved by a few solvers (Absil et al., 2009; Wen and Yin, 2013), here we employ the solver *Algorithm 2* in (Wen and Yin, 2013) to solve Equation 4.16, which is a more efficient optimization algorithm with code publicly available. Since this algorithm requires the derivative

of the objective function as one input, we obtain the derivative of Equation 4.16 with respect to $\mathbf{B}$:

$$\begin{aligned}\nabla_{\mathbf{B}}\mathscr{L}(\mathbf{B}) =&\mathbf{B}\mathbf{Q}^{\mathbf{T}}\mathbf{Q} - \mathbf{X}_{(1)}\mathbf{Q} + \mu(\mathbf{B} - \mathbf{P}) - \\ &\mathbf{U} + \alpha\left(\mathbf{L}\mathbf{B} + \mathbf{L}^{\mathbf{T}}\mathbf{B}\right)\end{aligned} \tag{4.17}$$

Then the auxiliary matrix $\mathbf{P}$ can be optimized by setting the derivative of Equation 4.16 with respect to $\mathbf{P}$ as 0. We have:

$$\mathbf{P} = \left(2\mathbf{X}_{(2)}\mathbf{O} + \mu\mathbf{B} - \mathbf{U}\right)\left(2\mathbf{O}^{\mathbf{T}}\mathbf{O} + \mu\mathbf{I}\right)^{-1} \tag{4.18}$$

where $\mathbf{O} = \mathbf{B} \odot \mathbf{A} \in \mathbf{R}^{(n*v)\times k}$ and $\mathbf{X}_{(2)} \in \mathbf{R}^{n\times(n*v)}$ is the mode-2 matricization of tensor $\mathcal{X}$.

After updating $\mathbf{B}$ and $\mathbf{P}$, we optimize the Lagrangian multipliers $\mathbf{U}$ by gradient ascent:

$$\mathbf{U} \leftarrow \mathbf{U} + \mu(\mathbf{P} - \mathbf{B}) \tag{4.19}$$

Note that in our experiment, we initialize μ as 10^{-6}, and set $\mu_{max} = 10^7$. Each time after $\mathbf{U}$ is updated, we adjust μ by $\mu = \min(\rho\mu, \mu_{max})$, where we set $\rho = 1.05$.

Update factor matrix A. Next, we fix $\mathbf{B}$ and optimize $\mathbf{A}$. Following Equation 4.12, the objective function with respect to $\mathbf{A}$ is:

$$\min_{\mathbf{A}} \left\|\mathbf{A}\mathbf{Z}^{\mathbf{T}} - \mathbf{X}_{(3)}\right\|_F^2 \tag{4.20}$$

where $\mathbf{Z} = \mathbf{B} \odot \mathbf{P} \in \mathbf{R}^{(n*n)\times k}$ and $\mathbf{X}_{(3)} \in \mathbf{R}^{v\times(n*n)}$ is the mode-3 matricization of $\mathcal{X}$, thus this can be solved directly.

By performing the above optimization steps iteratively until convergence, we can obtain the optimal indicator matrix $\mathbf{B}$ for the multi-view clustering stage, thus knowing which graphs are clustered together by performing k-means algorithm on the row vectors of $\mathbf{B}$. The resulted cluster information will be used for determining the weight matrix $\mathbf{W}$ in the multi-view graph embedding stage.

Now we solve the optimization problem in Equation 4.13 with respect to the weight matrix $\mathbf{W}$. According to the locally linear embedding approach proposed in (Saul and Roweis, 2000), such a minimization problem with respect to vectors can be solved as a constrained least squares problem. Since the Frobenius norm for matrices can be regarded as a generalization of the l_2 norm for vectors, we can directly derive the following equation based on the analysis in (Saul and Roweis, 2000):

$$\left\|\mathbf{X}_i - \sum_j w_{ij}\mathbf{X}_j\right\|_F^2 = \sum_{jr} w_{ij}w_{ir}\mathbf{C}_{jr} \tag{4.21}$$

where G_j and G_r are two neighbor graphs of G_i in the same cluster. $\mathbf{C}_{jr}$ is the local covariance matrix, and it can be computed by

$$\mathbf{C}_{jr} = \frac{1}{2}(M_j + M_r - m_{jr} - M_0) \tag{4.22}$$

where m_{jr} denotes the squared distance between the jth and rth neighbors of G_i, and we compute it based on both the original affinity matrices from v views and the graph embeddings of G_j and G_r by

$$m_{jr} = \frac{1}{2}(\frac{1}{d}\sum_{d=1}^{v}\left\|\mathbf{G}_j^{(d)} - \mathbf{G}_r^{(d)}\right\|_F^2) + \frac{1}{2}(\|\mathbf{F}_j - \mathbf{F}_r\|_F^2) \tag{4.23}$$

Algorithm 2 MCGE

Input: $\mathcal{X}$, $\{\mathcal{T}_1, \cdots, \mathcal{T}_n\}$, c, k, α, β
Output: **B**, **F**
Initialize **B** s.t. $\mathbf{B}_0^{\mathrm{T}}\mathbf{B}_0 = \mathbf{I}_k$;
Initialize $\mathbf{F}_i$ for $i = 1, 2, \cdots, n$ s.t. $\mathbf{F}_{i0}^{\mathrm{T}}\mathbf{F}_{i0} = \mathbf{I}_c$;
while not converge **do**
    Compute **W** according to Equation 4.24;
    for $i = 1 : n$ **do**
        $t \leftarrow 0$;
        while not converge **do**
            Compute $\mathbf{F}_{it+1}$ by solving Equation 4.8;
            $t \leftarrow t + 1$;
        end while
    end for
    Update **B** by solving Equation 4.12;
    Update **A** by solving Equation 4.20;
    Cluster **B** by k-means;
end while

$M_j = \sum_z m_{jz}$, $M_r = \sum_z m_{rz}$ and $M_0 = \sum_{jr} m_{jr}$. Then the optimal weights can be obtained by:

$$w_{ij} = \frac{\sum_r \mathbf{C}_{jr}^{-1}}{\sum_{lz} \mathbf{C}_{lz}^{-1}} \tag{4.24}$$

For details about the above derivation for the solution, readers can refer to the illustrations in (Saul and Roweis, 2000).

Once the weight matrix **W** is obtained, we can easily solve the optimization problem in Equation 4.8 following the same ADMM steps as shown above for solving Equation 4.12. The overall optimization algorithm of MCGE is summarized in Algorithm 2.

4.5 Experiments and Evaluation

In order to evaluate the performance of the proposed method for multi-view clustering of graphs, we test our framework on real fMRI and DTI brain network data for connectome analysis and compare with a few of state-of-the-art multi-view clustering methods.

4.5.1 Data Collection and Preprocessing

In this work, we use two real datasets as follows:

- *Human Immunodeficiency Virus Infection (HIV)*: This dataset is collected from the Chicago Early HIV Infection Study at Northwestern University(Ragin et al., 2012a). This clinical study involves 77 subjects, 56 of which are early HIV patients (positive) and the other 21 subjects are seronegative controls (negative). These two groups of subjects do not differ in demographic characteristics such as age, gender, racial composition and education level. This dataset contains both the functional magnetic resonance imaging (fMRI) and diffusion tensor imaging (DTI) for each subject, from which we can construct the fMRI and DTI brain networks.
- *Bipolar*: This dataset consists of the resting-state fMRI and DTI image data of 52 bipolar I subjects who are in euthymia and 45 healthy controls with matched age and gender (Cao et al., 2015c; Ma et al., 2017).

We perform preprocessing on the HIV dataset using the standard process as illustrated in (Cao et al., 2015a). First, we use the DPARSF toolbox[1] to process the fMRI data. We realign the images to the first volume, do the slice timing correction and normalization, and then use an 8-mm Gaussian kernel to smooth the image spatially. The band-pass filtering (0.01-0.08 Hz) and linear trend removing of the time series are also performed. We focus on the 116 anatomical volumes of interest (AVOI), each of which represents a specific brain region, and

[1]http://rfmri.org/DPARSF.

extract a sequence of responds from them. Finally, we construct a brain network with the 90 cerebral regions. Each node in the graph represents a brain region, and links are created based on the correlations between different brain regions. For the DTI data, we use FSL toolbox[1] for the preprocessing and then construct the brain networks. The preprocessing includes distortion correction, noise filtering, repetitive sampling from the distributions of principal diffusion directions for each voxel. We parcellate the DTI images into the 90 regions same with fMRI via the propagation of the Automated Anatomical Labeling (AAL) on each DTI image (Tzourio-Mazoyer et al., 2002).

For the Bipolar dataset, the brain networks were constructed using the CONN[2] toolbox (Whitfield-Gabrieli and Nieto-Castanon, 2012). The raw EPI images were first realigned and co-registered, after which we perform the normalization and smoothing. Then the confound effects from motion artifact, white matter, and CSF were regressed out of the signal. Finally, the brain networks were derived using the pairwise signal correlations based on the 82 labeled Freesurfer-generated cortical/subcortical gray matter regions.

4.5.2 Baselines and Metrics

We compare our MCGE framework with six other baseline methods for the multi-view clustering task on brain networks. To the best of our knowledge, our proposed framework is the first work that jointly performs multi-view graph embedding and multi-view clustering of

---

[1]http://fsl.fmrib.ox.ac.uk/fsl/fslwiki.

[2]http://www.nitrc.org/projects/conn

graph instances. Therefore, for the evaluation, we apply the following state-of-the-art multi-view clustering methods and adapt them to perform the multi-view clustering task here.

- **SingleBest** applies spectral clustering on each single view and reports the best performance among them.
- **SEC** is a single view spectral embedding clustering framework proposed in (Nie et al., 2011). It imposes a linearity regularization on the spectral clustering model and uses both local and global discriminative information for the embedding.
- **CoRegSc** is the co-regularized based multi-view spectral clustering framework proposed in (Kumar et al., 2011). The centroid based approach is applied for the multi-view clustering task.
- **MultiNMF** is the multi-view clustering method based on joint nonnegative matrix factorization proposed by (Liu et al., 2013). This method searches for a factorization that gives compatible clustering solutions across multiple views.
- **AMGL** is a recently proposed multi-view spectral learning framework (Nie et al., 2016) that can automatically learn an optimal weight for each graph without introducing additive parameters.
- **SCMV-3DT** is a tensor based multi-view clustering method recently proposed in (Yin et al., 2016). It uses t-product in third-order tensor space, and represents multi-view data by a t-linear combination with sparse and low-rank penalty based on the circular convolution.

- **MCGE** is the proposed multi-view clustering framework in this work, which jointly performs multi-view graph embedding and multi-view clustering of the graph instances.

There are three main parameters in our model, which include the α in objective function (Equation 4.12), the β in objective function (Equation 4.8), and the dimension c of the row vectors in the graph embedding. We apply the grid search to find the optimal values for the parameters. For details, we do grid search for α and β in $\{10^{-4}, 10^{-3}, \cdots 10^{4}\}$, and the optimal c is selected by the grid search from $\{2, 3, \cdots, 12\}$. For evaluation, since there are two possible labels of the brain network instances in both of the two datasets, we set the number of clusters k to be 2, and test how well our method can group the brain networks of subjects with disorders and those of normal controls into two different clusters.

For fair comparisons of the baseline methods, we employ Litekmeans (Cai, 2011) for all the k-means clustering step if it is needed in the implementation of the six methods listed above. We repeat clustering for 20 times with random initialization as k-means depends on initialization.

To evaluate the quality of the clusters produced by different approaches, we use *Accuracy* and *Normalized Mutual Information (NMI)* as the evaluation metrics. For each experiment, we repeat 50 times and report the mean value along with standard deviation (std) as the results.

4.5.3 Performance Evaluations

4.5.3.1 Clustering Accuracy and NMI

As shown in Table VI and Table VII, our MCGE framework performs the best in the multi-view clustering task on both of the two datasets in terms of accuracy and NMI. Among the

TABLE VI: Results on HIV dataset (mean ± std).

Methods	*Accuracy*	*NMI*
SingleBest	0.561 ± 0.010	0.104 ± 0.007
SEC	0.523 ± 0.012	0.092 ± 0.011
AMGL	0.563 ± 0.002	0.132 ± 0.008
SCMV-3DT	0.576 ± 0.013	0.123 ± 0.019
MultiNMF	0.613 ± 0.016	0.197 ± 0.021
CoRegSc	0.626 ± 0.020	0.254 ± 0.013
MCGE	$\mathbf{0.682 \pm 0.019}$	$\mathbf{0.390 \pm 0.015}$

TABLE VII: Results on Bipolar dataset (mean ± std).

Methods	*Accuracy*	*NMI*
SingleBest	0.553 ± 0.012	0.098 ± 0.006
SEC	0.536 ± 0.012	0.103 ± 0.009
AMGL	0.558 ± 0.026	0.101 ± 0.012
SCMV-3DT	0.585 ± 0.009	0.132 ± 0.010
MultiNMF	0.642 ± 0.011	0.192 ± 0.015
CoRegSc	0.619 ± 0.024	0.170 ± 0.008
MCGE	$\mathbf{0.703 \pm 0.013}$	$\mathbf{0.264 \pm 0.012}$

seven methods, the first two methods are single view clustering methods, both of which achieve lower accuracy and NMI compared with the multi-view methods. In particular, the lowest accuracy is from SEC, which is a single view clustering method applied here by concatenating the features of all the views. Although the SEC method considers both global structure and local structure of graphs, it does not distinguish the features from different views, which leads to

a poor performance in the multi-view clustering. The SingleBest achieves its best performance on the fMRI brain networks for both datasets, which means that the fMRI data provide more discriminative information for the SingleBest method. By comparing SingleBest with SEC, we can find that if the multiple views are combined improperly, it may perform even worse than only using information from a single view.

Among the multi-view clustering methods, CoRegSc and MultiNMF have quite good performance, though not as good as the proposed MCGE method. This is mainly because that they consider the interactions between different views via joint modeling with the multiple views, while the other two multi-view methods do not. Comparatively, CoRegSc achieves slightly better results than the MultiNMF method on HIV dataset and vice versa on the Bipolar dataset. Compared to the proposed MCGE method, the common property of the other four multi-view clustering methods is that the features they learn for each view are based on vector representations. However, for graph instances, the structural information could barely be preserved by such vector representations, which could be the underlying reason of why these methods could not outperform our MCGE method. Moreover, by using tensor technique to model the multi-view graph-graph affinity as illustrated in Equation 4.9, MCGE can not only encode the latent interaction across different views, but also capture the graph-specific features through the graph kernels. From Table VI and Table VII, we can see that, as another tensor-based multi-view method, the SCMV-3DT does not achieve compatible results to MCGE. The reason behind this might be that although SCMV-3DT models the data into third-order tensor, it does

not consider the local structure of graphs, making it less effective for the multi-view clustering of graphs.

4.5.3.2 MCGE for Connectome Analysis

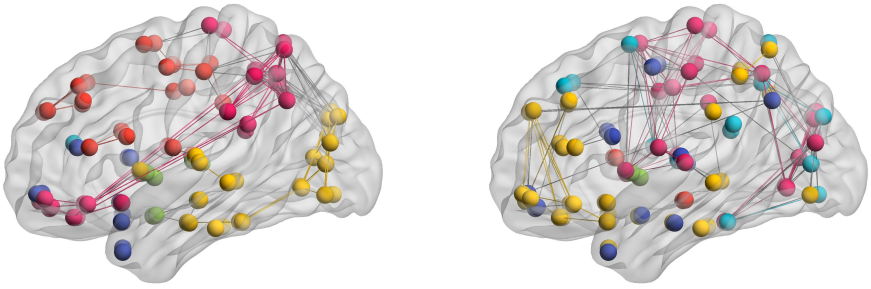
To evaluate the effectiveness of the proposed MCGE framework for connectome analysis, we investigate this approach for capturing the inner structure of connectomes in analysis of brain alterations induced by HIV infection and Bipolar affective disorder, respectively.

HIV is associated with heterogeneous changes in the brain and in cognitive function (Wang et al., 2011b). In many CNS(Central Nervous System) disorders, etiology is unknown. In contrast, HIV involves a known viral etiology. Therefore it is possible to study the brain in the early stages of injury. Studies of early HIV infection have found alterations in both structural and functional connectivity (Wang et al., 2011b). Moreover, a hallmark of HIV is neuroinflammation, which is a common characteristic of neurological injury from diverse causes, including traumatic, ischemic, developmental and neurodegenerative brain disorders. Since HIV infection is broadly relevant to many other neurological disorders, it represents an ideal model for evaluating the sensitivity of new frameworks for neuroimaging analysis.

We apply the proposed MCGE framework on the multi-view brain networks of the HIV dataset and obtain the clustering results as well as the multi-view graph embedding for each brain network. We further employ k-means algorithm (with $k = 6$) on the row vectors of the multi-view graph embedding for each brain network, and obtain the clustering relationship of their inner nodes, *i.e.*, the brain regions. Figure 14 shows an example of the resulting brain region clustering map of a normal control and that of an HIV patient. In this figure, each node

represents a brain region, and each edge indicates the correlation between two brain regions. Nodes of the same color represent the brain regions that are grouped into the same cluster by MCGE. As we can see from Figure 14, the clustering pattern of the HIV patient is quite different from the normal control. Nodes of the normal brain network are well grouped into several clusters, while nodes in the HIV brain network are less coherent. In addition, for the normal control, edges within each cluster are much more intense than the edges across different clusters. For example, in Figure 14(a), the pink nodes in the lower left and the pink nodes in the upper right are strongly connected with each other. While in Figure 14(b), the corresponding nodes in the lower left, which are mostly marked in yellow, have very few connections with those yellow nodes in the upper right. By looking into the connections, we can find that for the normal control, there are several pink nodes in the center of the brain which bridge the lower left part and the upper right part, while these intermediate nodes in the HIV brain are clustered in blue or pink instead of the same color (yellow) as the lower left part and the upper right part. This implies that the intermediate regions are probably injured so that they are no longer the bridges (or hubs) across other related regions. Some studies in neuroscience (Crossley et al., 2014) show that the highly-interconnected hub nodes are biologically costly due to higher blood flow or connection distances, and thus tend to be more sensitive to injury. Our observations in Figure 14 potentially reflect this evidence.

Then we apply the MCGE framework on the Bipolar dataset with the same steps as illustrated above for HIV dataset. The visualized results of a normal control and a bipolar subject are shown in Figure 15. Similarly to the observations above, as we can see from Figure 15,

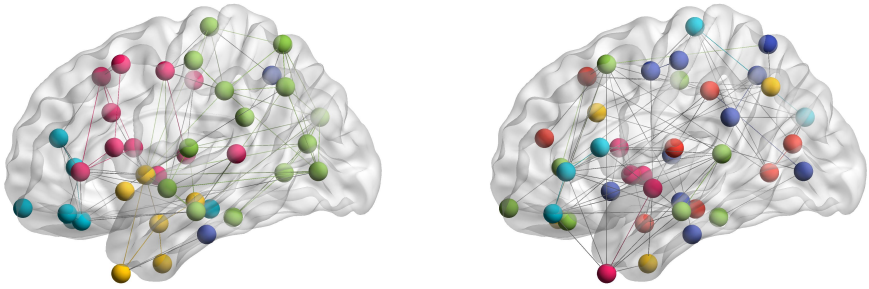


(a) normal control (b) HIV patient

Figure 14: Comparison of the connectomes captured from the brain networks of a normal control and an HIV patient

the cluster information of normal control is quite different from the bipolar subject. The connectomes of the normal control are well organized, while the corresponding nodes in the brain network of the bipolar subject spread out irregularly across different clusters. We can also find that for normal control, edges within each cluster are much more intense than the edges across different clusters, while this is less the case for bipolar subject. The reason behind this observation is probably that the collaborative activities of different brain regions of the bipolar subject are not organized in a proper order as those of normal controls are.

These findings indicate that our proposed MCGE framework can distinguish brain alterations in neurological disorders from healthy controls. It also yields new information and insights concerning network perturbations in brain injury and neuroinflammation for further investigation and interpretation.

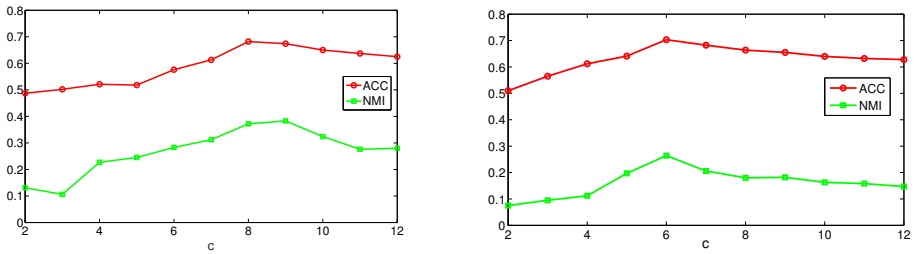


(a) normal control (b) bipolar subject

Figure 15: Comparison of the connectomes captured from the brain networks of a normal control and a bipolar subject

4.5.4 Parameter Sensitivity Analysis

In this section, we study the sensitivity of the proposed MCGE framework to the three parameters α, β, and c, and explore how the different values for parameters would affect the performance of MCGE in the multi-view clustering. We first look into the parameter c, which is the dimension of the row vectors in graph embedding. Figure 16 shows the multi-view clustering performance of MCGE on the two datasets with the c value varying from 2 to 12. From the figure, we can see that the value for c affects the performance of MCGE in both accuracy and NMI. The highest accuracy is achieved when c equals to 8 for HIV dataset and the best NMI occurs at 9. For Bipolar dataset, both the accuracy and NMI reach the peak when c equals to 6. The changing of accuracy and NMI with different c values has similar trend on the two datasets. With the increase of the c value, the performance first keeps rising up until it reaches the peak, and then it starts to decline. This changing trend is reasonable as when the dimension of graph

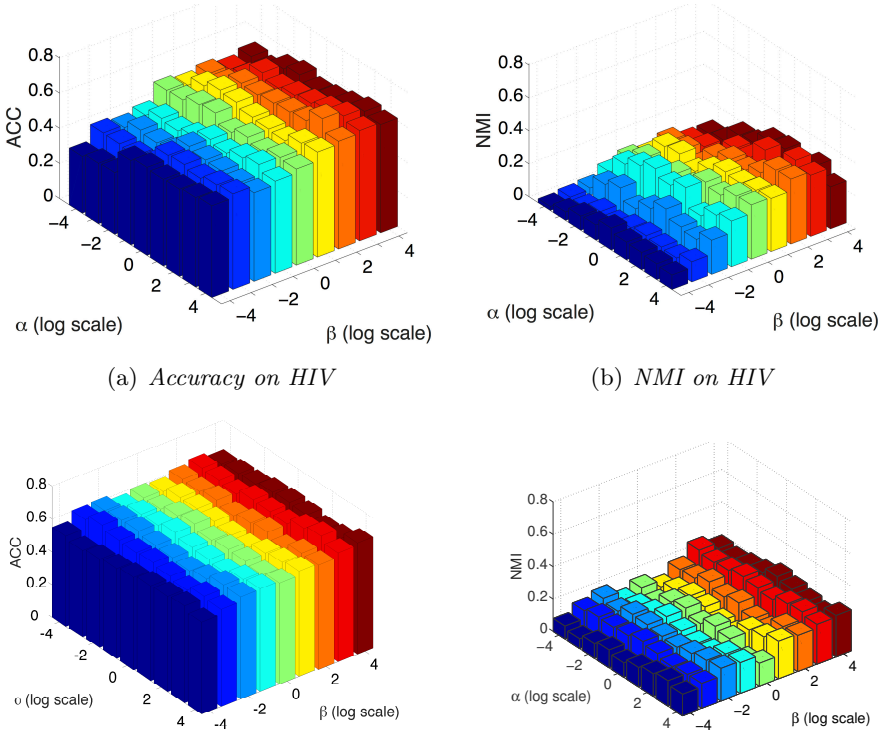


(a) *HIV* (b) *Bipolar*

Figure 16: *Accuracy* and *NMI* with different c

embedding is too small, it could not encode enough local structure information of the graph, resulting in poor performance for the clustering. When the dimension of graph embedding is set to be a large number, it may include much redundant information, making it less discriminative for the clustering task.

Now we evaluate the sensitivity of MCGE to α and β. As illustrated in Equation 4.12, α is the weight parameter which determines the extent that the local embedding structure is utilized for the multi-view clustering task. The higher the value for α, the more emphasis we put on the graph embedding for multi-view clustering modeling. Similarly, the parameter β balances how much influence the embedding of neighbor graphs would have on the multi-view graph embedding of each graph. For the evaluation, we set c to be 8 and run the MCGE framework with different values of α and β. The clustering accuracy and NMI achieved at different values of parameters for the two datasets are shown in Figure 17(a), 17(b), 17(c) and 17(d), respectively. As we can see from the figures, MCGE achieves different levels of accuracy and NMI when the values of α and β vary. The highest accuracy on HIV dataset is achieved when $\alpha = 10^3$, and



(a) *Accuracy on HIV* (b) *NMI on HIV*

(c) *Accuracy on Bipolar* (d) *NMI on Bipolar*

Figure 17: *Accuracy* and *NMI* with different α, β

$\beta = 10^2$, while the best NMI on HIV is achieved at $\alpha = 10^3$, and $\beta = 10^3$. On Bipolar dataset, both the highest accuracy and the best NMI are achieved when $\alpha = 10^3$, and $\beta = 10^3$. Notably, when the value for α is too small, both the accuracy and NMI achieved by MCGE are quite low, and the same situation holds for β. This is mainly because that if we set a small value to α, little information of graph embedding would be used for the multi-view clustering stage. Similarly, when β is too small, the graph embedding of neighbor graphs would hardly influence the multi-view graph embedding stage of each graph. On the other hand, when α and β are set

to be large values, the performance drops as well, as the influence imposed on those parts is too much. Therefore, finding an optimal combination of these parameter values is very important when applying MCGE framework for multi-view clustering.

4.6 Related Work

Our work relates to several branches of studies, which include multi-view clustering, graph embedding and connectome analysis.

Multi-view clustering is a clustering strategy for analyzing data with multiple views (Bickel and Scheffer, 2004) and it has been widely studied and applied in various domains (Shao et al., 2015; Shao et al., 2016; Lu et al., 2017). For example, the Canonical Correlation Analysis (CCA) based methods focus on constructing projections using multiple views(Bickel and Scheffer, 2004). In (Chaudhuri et al., 2009), a CCA based method is proposed and applied for audio-visual speaker clustering and hierarchical Wikipedia document clustering. Another main category of algorithms aim to integrate multiple views in the clustering process directly by optimizing the loss functions(Bickel and Scheffer, 2004). A typical work from this category is the co-regularized multi-view spectral clustering method proposed by (Kumar et al., 2011), which is also a baseline method used in our experiment. It performs multi-view clustering by co-regularizing the clustering hypotheses. In addition, matrix factorization based methods also form a category of multi-view clustering methods(Kalayeh et al., 2014; Liu et al., 2013), which use constraints to push multiple views towards consensus.

Graph embedding is a hot research topic in graph mining. The goal of graph embedding is to find low-dimensional representations of nodes in graphs that can preserve the important

structure and properties of graphs (Yan et al., 2007). It has drawn great interest from the data mining community, and has been extensively studied for various kinds of applications. In (Mousazadeh and Cohen, 2015), a new graph embedding algorithm is proposed based on Laplacian type operator on manifold, and it is applied for recovering the geometry of data and extending a function on new data points. Recently, a high-order proximity preserved embedding method is proposed in (Ou et al., 2016), where a general formulation that covers multiple high-order proximity measurements is first derived and a scalable embedding algorithm is then proposed to approximate the high-order proximity measurements.

Connectome analysis is a prominent emphasis area in the field of medical data mining. The "connectome", refers to the vast connectivity of neural systems at different levels involving both global and local structure information of the connections (Kaiser, 2011). Connectome analysis has been the focus of intense investigation owing to the tremendous potential to provide more comprehensive understanding of normal brain function and to yield new insights concerning many different brain disorders (Sporns et al., 2005; Cao et al., 2017; Ma et al., 2017). Most connectome analyses, however, aim to learn the structure from brain networks based on an individual neuroimaging modality (Cao et al., 2015c; Kuo et al., 2015; He et al., 2014b; He et al., 2017). For example, in (Cao et al., 2015c), the identification of discriminative subgraph patterns is studied on fMRI brain networks for bipolar affective disorder analysis. In (Ma et al., 2016), a multi-graph clustering method is proposed based on interior-node clustering for connectome analysis in fMRI resting-state networks. Although some recent work (Cahill et al., 2016) use multi-view brain networks in connectome analysis, they focus on the group-wise

functional community detection problem instead of doing multi-view clustering of the subjects. Here, we apply the proposed graph embedding based approach to facilitate the multi-view clustering of multiple brain networks simultaneously, thus providing a more comprehensive strategy for further neurological disorder identification.

CHAPTER 5

MULTI-VIEW GRAPH EMBEDDING WITH HUB DETECTION

(This chapter was partially published as "Multi-view Graph Embedding with Hub Detection for Brain Network Analysis", in ICDM'17 (Ma et al., 2017). DOI: http://doi.org/10.1109/ICDM.2017.123.)

5.1 Introduction

Recent years have witnessed an explosion of data in the form of graph representations. These data comes with a set of nodes and links between the nodes, for example the social networks with nodes representing users and links representing relationships among the users, and the brain networks with brain regions as nodes and the correlations among different regions as links. With the advanced capabilities for data acquisition, the links can usually be constructed from multiple sources or views of the data, which is usually called multi-view graph data. For instance, brain networks can be derived from fMRI (functional magnetic resonance imaging) and DTI (diffusion tensor imaging), which are two major neuroimaging techniques for brain data acquisition in neuroscience research and clinical applications. The fMRI brain networks reflect the correlations of different brain regions in functional activity, while the DTI networks encode the information of structural connections (*i.e.* white matter fiber paths) between different brain regions. Thus these two kinds of networks can serve as two views of the connectivity for brain network data (Ma et al., 2017).

Multi-view graph embedding, as a hot topic in multi-view graph learning, has drawn extensive attentions in the past decade. Most of the existing works in multi-view graph embedding aim to combine the information from all the views and obtain a lower dimensional but better feature representation of the nodes for the spectral clustering problem. For example, in (Kumar et al., 2011), a co-regularized multi-view spectral clustering method is proposed to find a consistent clustering across the multiple views. In (Papalexakis et al., 2013), two solutions based on minimum description length and tensor decomposition principles are proposed for graph clustering across multiple views. A multi-modal spectral clustering algorithm is presented in (Cai et al., 2011) to learn a commonly shared graph Laplacian matrix by unifying different views. In (Huang et al., 2012), an affinity aggregation spectral clustering algorithm is proposed, which seeks for an optimal combination of affinity matrices for the spectral clustering across multiple views. In (Li et al., 2015), a large-scale multi-view spectral clustering approach is proposed using local manifold fusion to integrate heterogeneous features of graphs.

Although these works introduced above can be used to obtain the graph embeddings from multiple views, none of them has considered the hubs when learning the multi-view graph embedding, making them less capable for the scenarios where hubs are also important for the clustering of nodes in graphs. The "hubs" refer to the bridging nodes that connect to different groups of nodes in a graph. For example, in a brain network, the hubs help bridge different groups of brain regions(van den Heuvel and Sporns, 2013), while in a social network, the hubs are known as "structural hole spanners"(He et al., 2016b), which refer to the users bridging different communities. The hubs in both of the two scenarios can potentially influence the node

clustering structure of the network, as they are the boundary-spanning nodes across different clusters and their neighbors usually spread out in different clusters. Therefore, hubs should be taken into account in the multi-view graph embedding learning process for achieving a clear and discriminative node clustering structure for the graph. Specifically, in neuroscience studies, the hubs of brain networks have been proven to be more biologically costly due to higher blood flow or connection distances, thus they tend to be more vulnerable to brain injuries (Crossley et al., 2014). As a result, the hubs will differ in the brain networks of normal people and those of the subjects with neurological disorder, which means the corresponding brain network embeddings of normal people and disordered subjects also tend to be different. Therefore, if we could consider the hubs when learning multi-view graph embeddings of brain networks, the resulted embeddings will be useful for distinguishing brain disordered subjects from normal controls.

In this work, we focus on jointly learning the multi-view graph embeddings and hubs for brain network analysis. There are three main challenges that must be addressed for this problem:

- As the task of multi-view graph embedding and the task of multi-view hub detection are naturally twisted, how to provide a joint learning framework such that both tasks can be solved at the same time and help improve the overall performance.
- It is often assumed that each individual view captures the partial information but they all admit the same underlying structure of the data. How to leverage the multi-view graph data for obtaining a good unified graph embedding across all the views?

- How to decide the importance of each view of the data when combining them for the multi-view learning task?

To address the above challenges, we propose an auto-weighted multi-view graph embedding with hub detection (MVGE-HD) framework. Our contributions can be summarized as follows:

- To the best of our knowledge, this is the first work to solve the problem of multi-view graph embedding with hub detection.
- The proposed MVGE-HD framework can jointly learn the multi-view graph embeddings and identify the hubs, instead of separating them into different steps. By considering the hubs, the obtained embeddings will reflect a clearer node clustering structure of the graph, which can better facilitate the further analysis of the graph.
- Our framework can automatically tune the importance of each view for the multi-view graph embedding with hub detection, avoiding the problem that might be caused otherwise by different parameter settings and thus having good generalization ability.
- We apply the MVGE-HD framework on two real brain network datasets (HIV and Bipolar) to investigate the multi-view brain region clustering structure and the hubs in brain networks for neurological disorder analysis, as a topic discussed for the first time in the literature of neuroscience study as well. The experimental results show the effectiveness of MVGE-HD for multi-view brain network analysis.

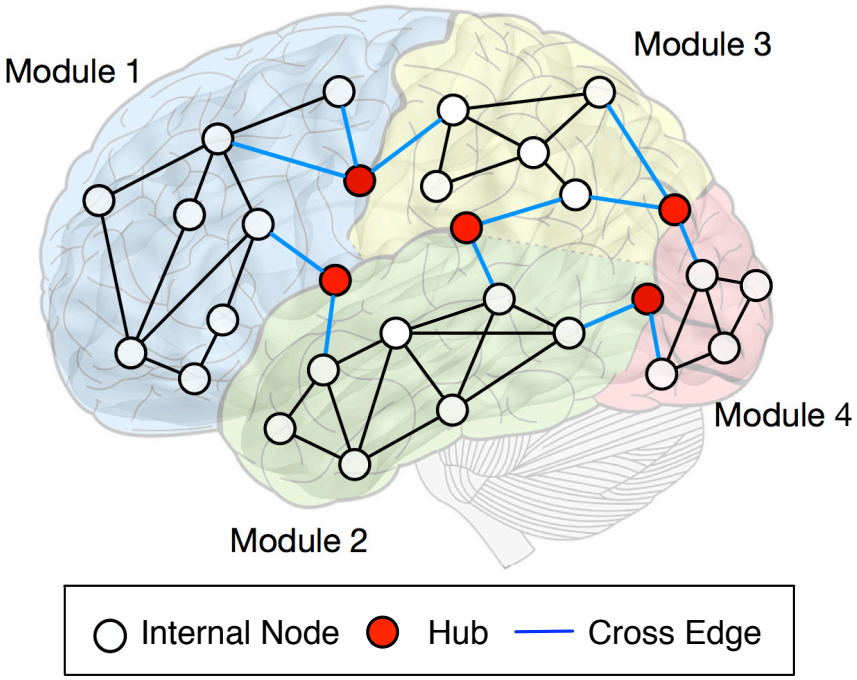


Figure 18: An brain network example with four modules and five hubs

5.2 Preliminaries

In this section, we introduce some notations and terminologies that we will use in this work. Then we establish some definitions and formulate the problem formally.

Notations. Vectors are denoted by boldface lowercase letters, and matrices are denoted by boldface capital letters. An element of a vector $\mathbf{x}$ is denoted by x_i, and an element of a matrix $\mathbf{X}$ is denoted by x_{ij}. For a matrix $\mathbf{X} \in \mathbb{R}^{n \times m}$, its i-th row and j-th column are denoted by $\mathbf{x}^i$ and $\mathbf{x}_j$, respectively. The Frobenius norm of $\mathbf{X}$ is defined as $\|\mathbf{X}\|_F = \sqrt{\sum_{i=1}^{n} \|\mathbf{x}^i\|_2^2}$, and the $\ell_{2,1}$ norm of $\mathbf{M}$ is defined as $\|\mathbf{M}\|_{2,1} = \sum_{i=1}^{n} \left\|\mathbf{m}^i\right\|_2$. For any vector $\mathbf{x} \in \mathbb{R}^n$, $Diag(\mathbf{x}) \in \mathbb{R}^{n \times n}$

is the diagonal matrix whose diagonal elements are x_i. $\mathbf{I}_n$ denotes an identity matrix with size n. We denote an undirected graph with m views as $G = (V, E_{(1)}, E_{(2)}, \cdots, E_{(m)})$, where V is the set of nodes and $E_{(i)} \subset V \times V$ is the set of edges from view i of G. We denote the affinity matrices of the multi-view graph G by $A = \{\mathbf{A}_{(1)}, \mathbf{A}_{(2)}, \cdots, \mathbf{A}_{(m)}\}$, where $\mathbf{A}_{(i)} \in \mathbb{R}^{n \times n}$ is the weighted affinity matrix in view i, and its entry denotes the pairwise affinity between nodes of G in view i.

We assume $\mathbf{F} \in \mathbf{R}^{n \times k}$ is an embedding of G, and then the i-th row vector of $\mathbf{F}$ (*i.e.*, $\mathbf{f}^i$) represents the embedding of node i. We call k the dimension of the embedding $\mathbf{F}$. If we run k-means algorithm on the set of row vectors of $\mathbf{F}$ and set the number of clusters as k, we will get a clustering assignment of the n nodes into k clusters. Thus an embedding of a graph usually implies its node clustering structure. We assume the k clusters are represented by $C = \{C_1, \cdots, C_k\}$, with $V = C_1 \cup \cdots \cup C_k$ and $C_i \cap C_j = \varnothing$ for every pair i, j with $i \neq j$. Based on these assumptions, we give the following definitions.

Definition 1. (Internal Node) For any node $v_i \in C_x$, if all the nodes that v_i have connections with belong to the same cluster C_x, node v_i is called an internal node.

Definition 2. (Hub) For any node $v_i \in C_x$, if there exists some neighboring node $v_j \in C_y (x \neq y)$, node v_i is called a hub.

Definition 3. (Cross Edge) For any edge $e_{ij} = (v_i, v_j) \in E$, if $v_i \in C_x$ and $v_j \in C_y (x \neq y)$, edge e_{ij} is called a cross edge.

Figure 18 shows an example of a brain network with four modules and five hubs. Note that in brain networks, the clusters of brain regions are often called "modules". In some works

of brain hub analysis, the hubs shown in Figure 18 are called "connector hubs" while another kind of hubs are called "provincial hubs" (van den Heuvel and Sporns, 2013), which refer to the internal node with high centrality within a module. In this work, the hubs we considered are the "connector hubs" stated in those works.

5.3 Methodology

In this section, we first present the proposed approach for multi-view graph embedding with hub detection. Then we derive the auto-weighted framework for the proposed approach.

5.3.1 Multi-view Graph Embedding with Hub Detection

Graph embedding, as an important tool in topological graph theory, has been widely studied for graph data analysis (Belkin and Niyogi, 2001; Fu and Ma, 2012; Yan et al., 2007). In the literature of graph embedding, hubs are seldom considered along with the embedding learning. However, in many graph learning scenarios, hubs play an important role for node clustering or graph embedding analysis. As shown in Figure 18, the hubs are those boundary-spanning nodes across different clusters in the graph, and their neighbors naturally occur in different clusters, and thus the hubs may blur the boundary between clusters. If we want to obtain a graph embedding that can encode a clear node clustering structure, it is crucial to enable the graph embedding approach to have the discriminative ability for such boundary-spanning nodes, *i.e.*, the hubs, and thus encoding only characterizing internal nodes in the graph. To solve this problem, the $\ell_{2,1}$-norm penalty is introduced to the context of node clustering and has been proven to be an effective strategy for dealing with the boundary-spanning nodes and improving the node clustering (He et al., 2016b; Ma et al., 2016). In this work, we employ

the similar strategy and incorporate it into our multi-view graph embedding and hub detection framework.

To derive our multi-view framework, we first formulate the problem of single-view graph embedding with hub detection. Given the affinity matrix $\mathbf{A}_{(v)}$ and the diagonal matrix $\mathbf{D}_{(v)}$ with $d_{(v)_{ii}} = \sum_{j=1}^{n} a_{ij}$ for view v of the graph G, we intend to obtain a graph embedding $\mathbf{F}_{(v)} \in \mathbf{R}^{n \times k}$. Based on the analysis in (Ma et al., 2016), the value of $\mathbf{f}_{(v)}^{i}$ at node v_i can be formulated as the weighted average of $\mathbf{f}_{(v)}^{i}$ at neighbors of v_i, where the weights are proportional to the edge weights in adjacency matrix $\mathbf{A}_{(v)}$, thus we can have the following objective function

$$\min_{\mathbf{F}_{(v)}} \left\| \mathbf{F}_{(v)} - \mathbf{D}_{(v)}^{-1} \mathbf{A}_{(v)} \mathbf{F}_{(v)} \right\|_F^2 \tag{5.1}$$

As discussed above, we need to make the embedding matrix $\mathbf{F}_{(v)}$ have discriminative ability for the hubs for inducing a clearer node clustering structure of G. Based on (He et al., 2016b) and (Ma et al., 2016), we apply the $\ell_{2,1}$-norm penalty and orthogonality constraints to promote the row-wise sparsity, so as to discriminate the hubs and encode only characterizing internal nodes. Then the problem of graph embedding with hub detection on single view becomes

$$\begin{aligned} \min_{\mathbf{F}_{(v)}} \ & \left\| \mathbf{F}_{(v)} - \mathbf{D}_{(v)}^{-1} \mathbf{A}_{(v)} \mathbf{F}_{(v)} \right\|_{2,1} \\ \text{s.t.} \ & \mathbf{F}_{(v)}^{\mathrm{T}} \mathbf{F}_{(v)} = \mathbf{I}_k \end{aligned} \tag{5.2}$$

For the multi-view graph learning task, we consider combining the information from the multiple views of graph G and obtaining a unified graph embedding across all the views, which

can better encode the embedding structure while considering the multi-view hubs as well. To achieve this goal, we propose to use the weighted combination of the graph embedding from each view, and we formulate it as follows.

We assume the unified embedding matrix across all the views of graph G is represented by $\mathbf{F} \in \mathbf{R}^{n \times k}$, where k is the dimension of the row vectors. Then the multi-view graph embedding with hub detection can be formulated as the following problem

$$\begin{aligned} \min_{\mathbf{F}} \ & \sum_{v=1}^{m} \alpha_{(v)} \left\| \mathbf{F} - \mathbf{D}_{(v)}^{-1} \mathbf{A}_{(v)} \mathbf{F} \right\|_{2,1} \\ \text{s.t.} \ & \mathbf{F}^{\mathrm{T}} \mathbf{F} = \mathbf{I}_k \end{aligned} \tag{5.3}$$

where $\alpha_{(v)}$ is the weight parameter for view v. Note that here the value of the weight parameter $\alpha_{(v)}$ is decided by an auto-tuning procedure, which will be introduced later in Section 5.3.2.

As the above minimization problem involving $\ell_{2,1}$ norm is nontrivial to solve directly, we further derive Equation 5.3 based on the following lemma (He et al., 2012b).

Lemma 2. Let $\phi(.)$ be a function satisfying the conditions: $x \to \phi(x)$ is convex on R; $x \to \phi(\sqrt{x})$ is convex on R_+; $\phi(x) = \phi(-x), \forall x \in R$; $\phi(x)$ is C^1 on R; $\phi''(0^+) \geq 0$, $\lim_{x \to \infty} \phi(x)/x^2 = 0$. Then for a fixed $\|\mathbf{u}^i\|_2$, there exists a dual potential function $\varphi(.)$, such that

$$\phi(\|\mathbf{u}^i\|_2) = \inf_{p \in R} \{p \|\mathbf{u}^i\|_2^2 + \varphi(p)\} \tag{5.4}$$

where p is determined by the minimizer function $\varphi(.)$ with respect to $\phi(.)$.

Let $\mathbf{P}_{(v)} = \mathbf{F} - \mathbf{D}_{(v)}^{-1}\mathbf{A}_{(v)}\mathbf{F}$. According to the analysis for the $\ell_{2,1}$ norm in (He et al., 2012b), if we define $\phi(x) = \sqrt{x^2 + \epsilon}$, we can replace $\|\mathbf{P}_{(v)}\|_{2,1}$ with $\sum_{j=1}^{n} \phi(\|\mathbf{p}_{(v)}^{j}\|_2)$. Thus, based on Lemma 2, we reformulate the objective function of Equation 5.3 as follows:

$$\begin{aligned} \min_{\mathbf{F}} \ & \sum_{v=1}^{m} \alpha_{(v)} \mathrm{Tr}\left(\mathbf{P}_{(v)}^{\mathrm{T}} \mathbf{Q}_{(v)} \mathbf{P}_{(v)}\right) \\ \text{s.t.} \ & \mathbf{F}^{\mathrm{T}}\mathbf{F} = \mathbf{I}_k \end{aligned} \tag{5.5}$$

where $\mathbf{Q}_{(v)} = Diag(\mathbf{q}_{(v)})$, and $\mathbf{q}_{(v)}$ is an auxiliary vector of the $\ell_{2,1}$ norm. The elements of $\mathbf{q}_{(v)}$ are computed by $q_{(v)_j} = \frac{1}{2\sqrt{\|\mathbf{p}_{(v)}^{j}\|_2^2 + \epsilon}}$, where ϵ is a smoothing term and is usually set to be a small constant value (we set $\epsilon = 10^{-4}$ in this work).

Plugging $\mathbf{P}_{(v)} = \mathbf{F} - \mathbf{D}_{(v)}^{-1}\mathbf{A}_{(v)}\mathbf{F}$ into Equation 5.5, we can have the full form of the objective function with respect to $\mathbf{F}$ as

$$\begin{aligned} \min_{\mathbf{F}} \ & \sum_{v=1}^{m} \alpha_{(v)} \mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}} \mathbf{L}_{(v)} \mathbf{F}\right) \\ \text{s.t.} \ & \mathbf{F}^{\mathrm{T}}\mathbf{F} = \mathbf{I}_k \end{aligned} \tag{5.6}$$

where $\mathbf{L}_{(v)} = \left(\mathbf{I}_n - \mathbf{D}_{(v)}^{-1}\mathbf{A}_{(v)}\right)^{\mathrm{T}} \mathbf{Q}_{(v)} \left(\mathbf{I}_n - \mathbf{D}_{(v)}^{-1}\mathbf{A}_{(v)}\right)$.

5.3.2 An Auto-weighted Framework: MVGE-HD

In the literature of multi-view graph learning, adding a weight parameter for each view tend to be a common way for balancing the influence of different views of the data, and the choice of the parameter values is usually crucial to the final performance(Cai et al., 2013; Li et al.,

2015; Xia et al., 2010). The optimal parameter value tends to change for different datasets. Therefore, it is critical to avoid this problem and make the multi-view graph embedding approach more general to be applied to different datasets. Inspired by the auto-weighted multiple graph learning strategy proposed in (Nie et al., 2016), we further derive our objective function and propose an auto-weighted framework called MVGE-HD as follows.

Following Equation 5.6, we assume there is no weight parameters explicitly defined for each view, and we take the following form for the general framework

$$\begin{aligned} &\min_{\mathbf{F}} \sum_{v=1}^{m} \sqrt{\mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}_{(v)}\mathbf{F}\right)} \\ &\text{s.t.} \quad \mathbf{F}^{\mathrm{T}}\mathbf{F} = \mathbf{I}_k \end{aligned} \tag{5.7}$$

The Lagrange function of Equation 5.7 can be written as

$$\min_{\mathbf{F}} \sum_{v=1}^{m} \sqrt{\mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}_{(v)}\mathbf{F}\right)} + \mathcal{G}\left(\mathbf{\Lambda}, \mathbf{F}\right) \tag{5.8}$$

where λ is the Lagrange multiplier, and $\mathcal{G}(\mathbf{\Lambda}, \mathbf{F})$ represents the Lagrange term derived from the constraint.

Then we take the derivative of Equation 5.8 with respect to $\mathbf{F}$ and set the derivative to be zero. We will have

$$\min_{\mathbf{F}} \sum_{v=1}^{m} \alpha_{(v)} \frac{\partial \mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}_{(v)}\mathbf{F}\right)}{\partial \mathbf{F}} + \frac{\partial \mathcal{G}\left(\mathbf{\Lambda}, \mathbf{F}\right)}{\partial \mathbf{F}} = 0 \tag{5.9}$$

where

$$\alpha_{(v)} = \frac{1}{2\sqrt{\mathrm{Tr}\left(\mathbf{F}^{\mathrm{T}}\mathbf{L}_{(v)}\mathbf{F}\right)}} \tag{5.10}$$

We can easily find that Equation 5.9 can be regarded as the solution to the problem in Equation 5.6 if $\alpha_{(v)}$ is set with a stationary value. However, as shown in Equation 5.10, the value of $\alpha_{(v)}$ depends on the variable $\mathbf{F}$. To solve this problem, we employ the alternating optimization scheme to update $\mathbf{F}$ and $\alpha_{(v)}$ alternately in an iterative manner. Given an initialized $\mathbf{F}$, we can compute the value for $\alpha_{(v)}$, according to Equation 5.10. Then the new $\alpha_{(v)}$ will be used consecutively to update $\mathbf{F}$ by solving Equation 5.6, so on and so forth until convergence. After this iterative optimization process, we will obtain both the learned weight $\alpha_{(v)}$ and the multi-view graph embedding $\mathbf{F}$ for Equation 5.6, which is the real problem we aim to solve.

In the above multi-view graph embedding problem, if view v can provide much useful information, we say it is a good view, and the value of $\mathrm{Tr}(\mathbf{F}^{\mathrm{T}}\mathbf{L}_{(v)}\mathbf{F})$ should be small. Based on Equation 5.10, the weight $\alpha_{(v)}$ will be large. Accordingly, a bad view will have a small weight. This indicates that the optimization scheme of the weights in our framework is reasonable.

Based on the above analysis, we can find that the proposed MVGE-HD framework can learn the multi-view graph embedding with hubs and the weight of each view simultaneously, thus can serve as a general framework for learning multi-view graph embedding on various datasets.

The details of the optimization process and the convergence analysis of the framework will be introduced later in Section 5.4.

5.4 Optimization

Following the analysis in Section 5.3, we present the iterative optimization process of MVGE-HD in this section. We start with the initialization of weight factor $\alpha_{(v)}$ for each view v, and set them to be $\frac{1}{m}$ equally. Now we compute $\mathbf{F}$ by solving the minimization problem (Equation 5.6). If we treat the $\sum_{v=1}^{m} \alpha_{(v)}\mathbf{L}_{(v)}$ in Equation 5.6 as a Laplacian matrix $\widetilde{\mathbf{L}}$, based on the spectral analysis in (Von Luxburg, 2007), the optimal $\mathbf{F}$ can be computed by solving the eigenvector problem of the matrix

$$\widetilde{\mathbf{L}} = \sum_{v=1}^{m} \alpha_{(v)} \left(\mathbf{I}_n - \mathbf{D}_{(v)}^{-1}\mathbf{A}_{(v)}\right)^{\mathrm{T}} \mathbf{Q}_{(v)} \left(\mathbf{I}_n - \mathbf{D}_{(v)}^{-1}\mathbf{A}_{(v)}\right) \tag{5.11}$$

Note that according to the illustration in Section 5.3.1, the diagonal matrix $\mathbf{Q}_{(v)}$ is dependent on $\mathbf{F}$. Therefore we need to compute $\mathbf{Q}_{(v)}$ first following its definition in Section 5.3.1 before updating $\mathbf{F}$. After we obtain the updated $\mathbf{F}$, we can use it to compute the weight factor $\alpha_{(v)}$ by Equation 5.10 for the next iteration, which will be used to compute $\mathbf{F}$ again following the same process discussed above. We summarize the overall optimization algorithm in Algorithm 1.

Based on the analysis in (Nie et al., 2016), it is obvious that the solution in Algorithm 3 will converge to a local optimum of the problem (Equation 5.7), as the updated $\mathbf{F}$ in each

Algorithm 3 MVGE-HD

Input: Affinity matrices for m views $A = \{\mathbf{A}_{(1)}, \mathbf{A}_{(2)}, \cdots, \mathbf{A}_{(m)}\}$; the dimension of the graph embedding k

Output: The graph embedding matrix $\mathbf{F}$,

1: Initialize $\mathbf{F}_0$ s.t. $\mathbf{F}_0^{\mathrm{T}}\mathbf{F}_0 = \mathbf{I}_k, t \leftarrow 0$;
2: **while** not converge **do**
3: Compute $\alpha_{(v)_t}$ for $v = 1, \cdots, m$ by Equation 5.10;
4: Set $\mathbf{Q}_{(v)_t} \leftarrow Diag(\frac{1}{2\sqrt{\|\mathbf{p}_{(v)}^j\|_2^2 + \epsilon}})$;
5: Compute $\mathbf{F}_{t+1}$ by calculating the eigenvectors corresponding to the 2nd to $(k+1)$-th smallest eigenvalues of matrix $\widetilde{\mathbf{L}}$ in Equation 5.11;
6: $t \leftarrow t + 1$;
7: **end while**

iteration of Algorithm 3 monotonically decrease the objective function in Equation 5.7. For details about the theorem and proof, users can refer to the illustrations in (Nie et al., 2016).

5.5 Experiments and Analysis

5.5.1 Data Collection and Preprocessing

In this work, we use two real datasets as follows:

- *Human Immunodeficiency Virus Infection (HIV)*: This dataset is collected from the Chicago Early HIV Infection Study at Northwestern University(Ragin et al., 2012a). This clinical study involves 77 subjects, 56 of which are early HIV patients (positive) and the other 21 subjects are seronegative controls (negative). These two groups of subjects do not differ in demographic characteristics such as age, gender, racial composition and education level. This dataset contains both the functional magnetic resonance imaging (fMRI) and

diffusion tensor imaging (DTI) for each subject, from which we can construct the fMRI and DTI brain networks. Then we can treat them as graphs with two views.

- *Bipolar*: This dataset consists of the fMRI and DTI image data of 52 bipolar I subjects who are in euthymia and 45 healthy controls with matched age and gender. The resting-state fMRI scan was acquired on a 3T Siemens Trio scanner using a T2*-weighted echo planar imaging (EPI) gradient-echo pulse sequence with integrated parallel acquisition technique (IPAT), set with TR = 2 sec, TE = 25 msec, flip angle = 78, matrix = 64x64, FOV = 192 mm, in-plane voxel size = 3x3 mm, slice thickness = 3 mm, 0.75 mm gap, and 30 total interleaved slices. Two TRs at the beginning of the scan were discarded to allow for scanner equilibration. There are 208 volumes acquired for the total sequence time of 7 min and 2 sec. Diffusion weighted MRI data were acquired on a Siemens 3T Trio scanner. 60 contiguous axial brain slices were collected using the following parameters: 64 diffusion-weighted (b = 1000s/mm^2) and 1 non-diffusion weighted scan; field of view (FOV) 190x190 mm; voxel size 2x2x2 mm; TR = 8400 ms; TE = 93 ms. In addition, high-resolution structural images were acquired using T1-weighted magnetization-prepared rapid-acquisition gradient echo (MPRAGE; FOV 250x250 mm; voxel size: 1x1x1 mm; TR = 1900 ms, TE = 2.26 ms, flip angle = 9 °) (Leow et al., 2013).

We perform preprocessing on the HIV dataset using the standard process as illustrated in (Cao et al., 2015a). First, we use the DPARSF toolbox[1] to process the fMRI data. We realign

[1]http://rfmri.org/DPARSF.

the images to the first volume, do the slice timing correction and normalization, and then use an 8-mm Gaussian kernel to smooth the image spatially. The band-pass filtering (0.01-0.08 Hz) and linear trend removing of the time series are also performed. We focus on the 116 anatomical volumes of interest (AVOI), each of which represents a specific brain region, and extract a sequence of responds from them. Finally, we construct a brain network with the 90 cerebral regions. Each node in the graph represents a brain region, and links are created based on the correlations between different brain regions. For the DTI data, we use FSL toolbox[1] for the preprocessing and then construct the brain networks. The preprocessing includes distortion correction, noise filtering, repetitive sampling from the distributions of principal diffusion directions for each voxel. We parcellate the DTI images into the 90 regions same with fMRI via the propagation of the Automated Anatomical Labeling (AAL) on each DTI image (Tzourio-Mazoyer et al., 2002).

For the Bipolar dataset, the brain networks were constructed using the CONN[2] toolbox (Whitfield-Gabrieli and Nieto-Castanon, 2012). The raw EPI images were first realigned and co-registered, after which we perform the normalization and smoothing. Then the confound effects from motion artifact, white matter, and CSF were regressed out of the signal. Finally, the brain networks were derived using the pairwise signal correlations based on the 82 labeled Freesurfer-generated cortical/subcortical gray matter regions.

---

[1]http://fsl.fmrib.ox.ac.uk/fsl/fslwiki.

[2]http://www.nitrc.org/projects/conn

5.5.2 Baselines and Evaluation Metrics

In brain network study, an important task is to use the graph connectivity features for neurological disorder analysis. As introduced above, both the HIV dataset and Bipolar dataset have the two-view brain networks of a group of subjects with neurological disorder and a group of normal controls. In this work, to evaluate the effectiveness of the proposed MVGE-HD framework for brain network analysis, we apply MVGE-HD on each of the multi-view brain network instances in HIV dataset and Bipolar dataset, and then we use the learned multi-view graph embedding as the feature of each instance and use it for clustering the subjects in HIV dataset and Bipolar dataset, respectively. Then we evaluate the MVGE-HD approach by investigating how well the resulting multi-view graph embedding of MVGE-HD can help in separating the neurological disordered subjects and normal controls. In addition, we also look into the hubs learned by our framework and analyze them in the perspective of neuroscience.

We compared our MVGE-HD framework with seven other baseline methods on the HIV and Bipolar datasets. As our proposed framework is the first work on jointly learning multi-view graph embedding and hubs, there is no other existing method proposed for the same problem. Therefore, for the evaluation, we apply several state-of-the-art methods of multi-view graph embedding as baselines and adapt them for the problem here.

- **SingleBest** applies the single-view version of the proposed MVGE-HD framework (*i.e.*, Equation 5.2) on each single view and reports the best performance among them.

- **SEC** is a single view spectral embedding clustering approach proposed in (Nie et al., 2011). It imposes a linearity regularization on the spectral clustering model and uses both local and global discriminative information for the embedding.
- **CoRegSc** is the co-regularized based multi-view spectral clustering framework proposed in (Kumar et al., 2011). The centroid based approach is applied for the multi-view graph embedding task here.
- **MMSC** is the multi-modal spectral clustering method proposed by (Cai et al., 2011). It aims to learn a commonly shared graph Laplacian matrix by unifying different views.
- **AMGL** is a recently proposed multi-view spectral learning approach (Nie et al., 2016) that can automatically learn an optimal weight for each graph without introducing additive parameters.
- **BC+CoRegSc** is the method we combined with Betweenness Centrality (Brandes, 2001) and CoRegSc for evaluating if the hubs detected would help improve the multi-view graph embedding of CoRegSc, and also for comparing with our method. Betweenness Centrality (BC) is a popular method for hub detection in both social network and brain network. We first apply BC on each view of the data to obtain the top-k hubs, and then we remove their connections with other nodes in the graph by setting the corresponding values in affinity matrix to be 0. Then we run CoRegSc with the new affinity matrices from all the views for learning the multi-view graph embedding.
- **MVGE-HD*** represents the proposed approach in Equation 5.3 without auto-weighted ability. We set the weight parameter $\alpha_{(v)}$ as 0.5 for each of the two views, and evalu-

ate the performance for the comparison with the auto-weighted version of the proposed framework.

- **MVGE-HD** is the proposed auto-weighted framework for multi-view graph embedding with hub detection.

After we run each of the above algorithms on the data, we will obtain a multi-view graph embedding matrix $\mathbf{F}$ for each multi-view brain network instance. To facilitate the clustering of the instances, we use the following equation to compute the similarity between each pair of instances (Frey and Dueck, 2007).

$$s_{ij} = -\sqrt{\mathrm{Tr}\left(\left(\mathbf{F}_i - \mathbf{F}_j\right)^{\mathrm{T}}\left(\mathbf{F}_i - \mathbf{F}_j\right)\right)} \tag{5.12}$$

where $\mathbf{F}_i$ is the multi-view graph embedding of instance i and $\mathbf{F}_j$ is the multi-view graph embedding of instance j.

Then we apply the standard spectral clustering procedure (Shi and Malik, 2000) for the clustering of the brain network instances. For the k-means clustering step in the experiment, we use the Litekmeans (Cai, 2011) implementation.

As the weight factor $\alpha_{(v)}$ in the proposed MVGE-HD framework is auto-tuned, for fair comparisons of the baseline methods, we tune parameters for each of the baseline methods, and report their performance with the optimal parameter settings. The optimal value for the multi-view graph embedding dimension k is selected by the grid search from $\{5, 6, \cdots, 15\}$. For each experiment, we repeat 20 times and report the mean value with the standard deviation

(std) as the results. In the clustering stage of the brain network instances, we set the number of clusters to be 2, as there are two possible labels (*i.e.*, patient and normal control) in the HIV and Bipolar datasets.

We adopt the following measures for the evaluation.

- *Accuracy (ACC)*. Let c_i represent the clustering label result of the clustering algorithm and y_i represent the ground truth label of the two-view brain network instance i. Then *Accuracy* is defined as:

$$Accuracy = \frac{\sum_{i=1}^{n} \delta(y_i, map(c_i))}{n} \tag{5.13}$$

  where δ is the Kronecker delta function, and $map(c_i)$ is the best mapping function that permutes clustering labels to match the ground truth labels using the KuhnMunkres algorithm (Kuhn, 1955). A larger ACC indicates better clustering performance.

- *Normalized Mutual Information (NMI)*. Normalized Mutual Information is a measure used to evaluate the mutual information entropy between the resulted cluster labels and the real labels. The NMI for any two variables X and Y is defined as:

$$NMI = \frac{I(X,Y)}{\sqrt{H(X)H(Y)}} \tag{5.14}$$

  where $I(X,Y)$ computes the mutual information between X and Y. $H(X)$ and $H(Y)$ represent the entropies of X and Y, respectively. The larger the NMI value, the better the clustering performance.

TABLE VIII: Results on HIV dataset (mean ± std).

Methods	*ACC*	*NMI*
SingleBest	0.579 ± 0.011	0.086 ± 0.009
SEC	0.552 ± 0.010	0.058 ± 0.011
AMGL	0.582 ± 0.002	0.091 ± 0.006
MMSC	0.586 ± 0.013	0.105 ± 0.010
CoRegSc	0.625 ± 0.012	0.163 ± 0.015
BC+CoRegSc	0.635 ± 0.009	0.190 ± 0.008
MVGE-HD*	0.613 ± 0.010	0.152 ± 0.008
MVGE-HD	$\mathbf{0.701 \pm 0.012}$	$\mathbf{0.261 \pm 0.011}$

TABLE IX: Results on Bipolar dataset (mean ± std).

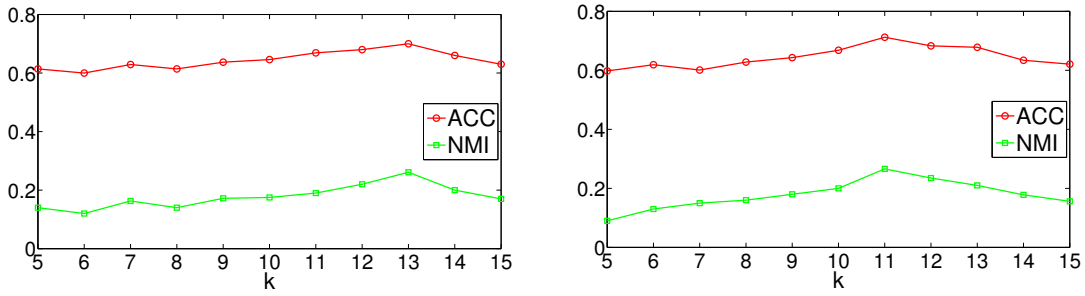
Methods	*ACC*	*NMI*
SingleBest	0.565 ± 0.012	0.074 ± 0.009
SEC	0.549 ± 0.012	0.067 ± 0.008
AMGL	0.563 ± 0.001	0.088 ± 0.006
MMSC	0.608 ± 0.014	0.119 ± 0.011
CoRegSc	0.637 ± 0.011	0.194 ± 0.013
BC+CoRegSc	0.641 ± 0.012	0.203 ± 0.009
MVGE-HD*	0.628 ± 0.010	0.175 ± 0.009
MVGE-HD	$\mathbf{0.712 \pm 0.010}$	$\mathbf{0.266 \pm 0.011}$

5.5.3 Performance Analysis

Table VIII and Table IX show the the clustering performance by using the multi-view graph embedding obtained with each of the seven methods on the HIV dataset and Bipolar dataset, respectively. As we can see from Table VIII and Table IX, the multi-view graph embedding obtained by the proposed MVGE-HD framework results in the best clustering performance on both of the two datasets in terms of *accuracy* and *NMI*. Among the eight methods, the Sin-

gleBest and SEC are the only two single-view graph embedding methods, and we can find that they both achieve lower accuracy compared with most of the multi-view methods, although the SingleBest performs slightly better than AMGL on Bipolar dataset in terms of *accuracy*. This indicates that the information combined from multiple views can lead to a better graph embedding result than that of a single view. Comparing with SEC, the SingleBest method achieves higher *accuracy* and *NMI* on both datasets. This is probably because that the SingleBest considers the hubs when doing graph embedding, while SEC only focuses on the spectral analysis for the embedding. In the experiment, the best performance of SingleBest and SEC both occur in the fMRI brain networks, which means that fMRI data provide more discriminative information for SingleBest and SEC.

Among the six multi-view graph embedding methods, the BC+CoRegSc, MVGE-HD* and the MVGE-HD consider the hubs when performing the multi-view graph embedding, while the three other methods do not. We can see that all the three methods that consider hub detection achieve better performance than the other three methods. This implies that detecting the hubs and reducing their effect in the multi-view graph embedding process benefit the task, and the multi-view graph embedding obtained in this case tend to be more discriminative for the analysis of multiple graph instances. Meanwhile, we can see that, although the BC+CoRegSc method performs better than the other baselines, the accuracy it achieves is still much lower than that of our proposed MVGE-HD approach. This is mainly due to the fact that the hub detection stage and multi-view graph embedding stage is separately done by BC+CoRegSc. The hubs detected by BC may not correspond to the hubs implied by the multi-view graph embedding



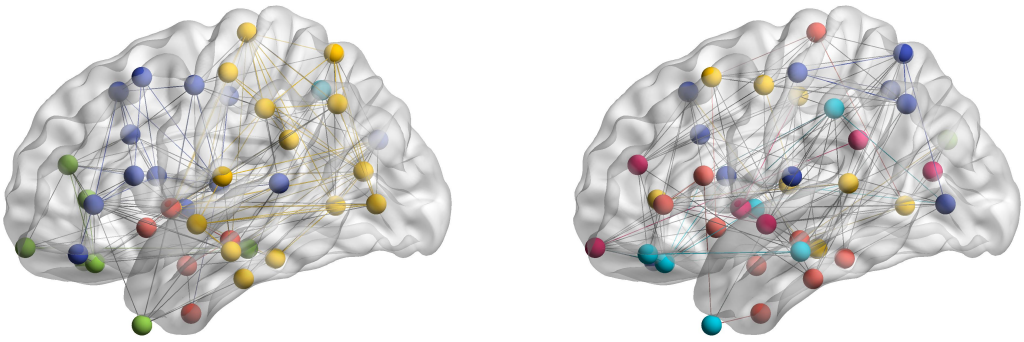
(a) *HIV* (b) *Bipolar*

Figure 19: *Accuracy* and *NMI* with different c

derived from CoRegSc, although some of the hubs detected may be helpful for the multi-view graph embedding stage by CoRegSc. Comparatively, in the proposed MVGE-HD framework, the hub detection is done along with the multi-view graph embedding, and by shrinking the embedding row vector of the potential hubs to zero, the resulted multi-view graph embedding would reflect a more discriminative node clustering structure of the graph. In addition, we find that the MVGE-HD* method, which is the version of MVGE-HD with an equal weight factor as 0.5 for each view, achieves much lower *accuracy* and *NMI* compared to the auto-weighted MVGE-HD framework. This indicates that the auto-weighted ability is very important for the multi-view graph embedding with hub detection task. In the multi-view learning process, different views may exert different levels of influence on the multi-view task, and the optimal weight for each view often varies from dataset to dataset. Therefore, the auto-weighted ability of the proposed MVGE-HD framework enables it be easily applied for different datasets.

In the proposed MVGE-HD framework, the only parameter is the dimension of multi-view graph embedding, which is the k introduced earlier. Now we evaluate the sensitivity of MVGE-HD to different values of k. Figure 19(a) and Figure 19(b) show the performance of MVGE-HD corresponding to the k values ranging from 5 to 15 on the HIV dataset and Bipolar dataset, respectively. As we can see from the figures, the value of k affects the performance in both *accuracy* and *NMI*. For the HIV dataset, the best *accuracy* and *NMI* are achieved when $k = 13$, while the best *accuracy* and *NMI* occurs at $k = 11$ for the Bipolar dataset. The changing of *accuracy* and *NMI* with respect to different k values have similar trend on both of the two datasets. With the increase of k value, the performance first keeps rising up until it reaches the peak, and then it starts to decline. This indicates that when the dimension of multi-view graph embdding is too low, it could not capture enough structure information of the graph, leading to poor performance. When the dimension is set to be a large value, it may contain much redundant information, thus being less discriminative to be used for the clustering task. Therefore, the dimension of the multi-view graph embedding for the MVGE-HD framework should be set based on the application scenarios.

To evaluate the effectiveness of the proposed MVGE-HD framework for brain region clustering analysis, after we obtain the multi-view graph embedding of all the brain networks, we further apply the k-means algorithm with k equal to the dimension value k on the row vectors of the multi-view graph embedding for each brain network instance and then we visualize the clustering results using the Brain Net Viewer toolbox (Xia et al., 2013). Figure 20 shows an example of the resulted visualized brain network with 6 clusters (*i.e.*, $k = 6$) of a normal control



(a) normal control (b) bipolar subject

Figure 20: Comparison of the brain region clusters resulted from MVGE-HD on the brain networks of a normal control and a bipolar subject

and a bipolar subject. In the figures, each node represents a brain region in the network, and the nodes with the same color refer to the brain regions that have been clustered into the same group, and the edges represent the connections between different brain regions.

As we can see from Figure 20(a), the clusters in the brain network of the normal control look quite clear, while the clusters in brain network of the bipolar subject as shown in Figure 20(b) is very messy. This indicates that the collaborations of different brain regions are well-organized for the normal control, as the regions close to each other in the brain are usually highly correlated and tend to collaborate more in brain activities. However, for the bipolar subject, the collaborations of the brain regions are probably in some kind of disorder, leading to the messy clusters as shown in Figure 20(b). Moreover, the big difference between the clustering maps of the two networks is probably partially due to the difference of their hubs.

Since MVGE-HD can detect the multi-view hubs and adjust the multi-view embedding with the hubs, when the hubs of the neurological disordered brain networks are different from those of normal people, the multi-view graph embedding guided by the hub detection of MVGE-HD would also be different for them. These observations coincide well with the findings about hubs in neuroscience study (Crossley et al., 2014). In addition, from Figure 20(a), we can also find that although some boundary nodes between the clusters have quite a few cross edges, which means they are the hubs in the brain network, the clusters resulted from MVGE-HD are not blurred by these nodes. This implies that our MVGE-HD approach can reduce the influence of these hubs, thus leading to clear cluster boundaries and discriminative clustering structure for the brain networks.

5.6 Related Work

Our work relates to several branches of studies, which include multi-view graph learning, hub detection and brain network analysis.

5.6.1 Multi-View Graph Embedding

Multi-view graph embedding has been a widely studied topic for the multi-view learning community in recent years. The key issue in multi-view graph embedding is how to combine the multiple views, so that both the consensus and complementary information across different views can be utilized for learning the embedding. The existing methods in this field can be divided into three categories. In the first category, the multiple views are often combined via integrating the affinity matrix or other graph features of each view. For example, in (Huang et al., 2012), a multi-view spectral clustering algorithm is proposed based on affinity aggregation,

which seeks for an optimal combination of affinity matrices for the spectral clustering across multiple views. In (Li et al., 2015), a large-scale multi-view spectral clustering method is introduced, where local manifold fusion is used for integrating heterogeneous information of graphs. The second category of works aim to learn a new Laplacian matrix by combining the Laplacian matrices of different views. For instance, a multi-modal spectral clustering algorithm is presented in (Cai et al., 2011) to learn a commonly shared graph Laplacian matrix by unifying different views. For the works in the third category, they aim to obtain a consistent clustering across all the views by adjusting the clustering along with learning features from the multiple views. In (Papalexakis et al., 2013), two solutions of multi-view graph embedding are proposed, which use the minimum description length and tensor decomposition principles respectively for graph clustering across multiple views. Another classic method for finding a consistent clustering across the multiple views is the co-regularized multi-view spectral clustering method proposed in (Kumar et al., 2011), which is also a baseline method we use in the experiments.

5.6.2 Hub Detection

Hub detection is also an essential research topic in graph mining. In the past decade, quite a few of works have been done in this area. Some of them focus on the structural hole detection problem for social network analysis (He et al., 2016b; Rezvani et al., 2015). In (Rezvani et al., 2015), they propose a method based on bounded inverse closeness centrality for analyzing the structural hole spanners, which are viewed as the vertices that can result in the maximum increase on the mean distance of the network if they are removed. In (He et al., 2016b), a model called HAM is proposed for jointly detecting the communities and structural holes in

social networks. They show that by removing the detected structural hole spanners, the quality of the learned communities can be improved. Some other works aim to use the hub detection measures for neuroscience study. For example, a review of network hubs in human brain is presented in (van den Heuvel and Sporns, 2013), and the rich-club organization of the human connectome is studied in (Van Den Heuvel and Sporns, 2011), which illustrate the important role that hubs play in human brains.

5.6.3 Brain Network Analysis

Brain network analysis is a prominent emphasis area in the field of medical data mining. So far, the researchers in this field aim to study the connectivity of neural systems at different levels involving both global and local structure information of the connections (Kaiser, 2011). Brain network analysis has been the focus of intense investigation owing to the tremendous potential to provide more comprehensive understanding of normal brain function and to yield new insights concerning many different brain disorders (Sporns et al., 2005; Cao et al., 2017; Ma et al., 2016). Most connectome analyses, however, aim to learn the structure from brain networks based on an individual neuroimaging modality (Cao et al., 2015c; Kuo et al., 2015). For example, in (Cao et al., 2015c), the identification of discriminative subgraph patterns is studied on fMRI brain networks for bipolar affective disorder analysis. In (Ma et al., 2016), a multi-graph clustering method is proposed based on interior-node clustering for connectome analysis in fMRI resting-state networks. Although some recent work (Cahill et al., 2016) use multi-view brain networks in connectome analysis, they focus on the group-wise functional community detection problem instead of doing multi-view graph embedding of each subject. Here, we apply the proposed

multi-view graph embedding on each subject, which further facilitates the clustering of all the subjects, thus providing a more comprehensive strategy for further neurological disorder identification.

CHAPTER 6

CONCLUSION

(Part of the chapter was previously published in (Ma et al., 2016; Ma et al., 2016; Ma et al., 2017; Ma et al., 2017).)

In this dissertation, we have explored the problem of learning from brain data for neurological disorder analysis. Towards this direction, we thoroughly studied four different research problems: spatio-temporal tensor analysis, multi-graph clustering, multi-view clustering with graph embedding and multi-view graph embedding with hub detection. We have evaluated the effectiveness of the proposed approaches in neurological disorder analysis by extensive experiments on various real brain imaging datasets. The main contributions of our works are summarized as follows:

- We have studied the problem of spatio-temporal tensor analysis for whole-brain fMRI classification. We have proposed a spatio-temporal tensor kernel (STTK) modeling method for the classification task. Different from conventional kernel methods, our approach exploits the inherent spatio-temporal structure to facilitate kernel learning, while considering both the correlation and discrepancy between the spatial domain and the temporal domain. STTK consists of three steps: (1) volumetric time series extraction for extracting discriminate information from the time series, (2) spatio-temporal feature extraction for obtaining a more compact and informative representation, and (3) tensor structure

mapping for kernel generation. Empirical studies on real-world fMRI brain images have demonstrated that our approach can significantly boost the fMRI classification performance in three different kinds of brain disorders (*i.e.*, Alzheimer's disease, ADHD and HIV).

- We have described and studied the problem of multi-graph clustering based on interior-node topology. To capture the local topological structure of the graphs, we perform the sparsity-inducing interior-node clustering on each graph. An iterative framework MGCT was proposed for multi-graph clustering based on interior-node topology of graphs. In this framework, the interior-node clustering and the multi-graph clustering are performed alternatively, where the results of interior-node clustering are exerted on multi-graph clustering and the multi-graph clustering in turn improves the interior-node clustering of each graph. After this iterative mutual reinforcement process, we can obtain a refined multi-graph clustering result, which can be used for further analysis of the graphs. Experiments on two real brain network datasets have demonstrated the superior performance of the proposed model for brain network clustering analysis.

- We have proposed MCGE, a Multi-view Clustering framework with Graph Embedding, to solve multi-view clustering problem on graph instances. MCGE first models the multi-view graph data as tensors and then learns the multi-view graph embeddings via tensor factorization. We further incorporate multi-view graph embedding into an iterative multi-view clustering framework, jointly performing multi-view clustering and graph embedding simultaneously. The results of multi-view clustering are used to refine the multi-view

graph embeddings, in turn, the updated multi-view graph embedding results are used to improve the multi-view clustering. By updating the clustering results and graph embeddings iteratively, the proposed MCGE framework will result in a better multi-view clustering solution. We have successfully applied our MCGE framework for unsupervised multi-view connectome analysis on HIV-induced brain alterations and bipolar affective disorder.

- We have presented an auto-weighted framework of Multi-view Graph Embedding with Hub Detection (MVGE-HD) for brain network analysis. We incorporate the hub detection task into the multi-view graph embedding framework so that the two tasks could benefit from each other. The MVGE-HD framework learns a unified graph embedding across all the views while reducing the potential influence of the hubs on blurring the boundaries between node clusters in the graph, thus leading to a clear and discriminative node clustering structure for the graph. With this approach, the multi-view embedding and hub detection on brain networks can be jointly performed. In addition, the proposed framework can automatically tune the importance of each view, avoiding the problem that might be caused otherwise by different parameter settings and thus having good generalization ability.

APPENDICES

.1 SDM Copyright Letter

In order for SIAM to include your paper in the 2016 SIAM International Conference on Data Mining proceedings, the following Copyright Transfer Agreement must be agreed to during the paper upload process.

Title of Paper:

Author(s):

COPYRIGHT TRANSFER AGREEMENT

Copyright to this paper is hereby irrevocably assigned to SIAM for publication in the **2016 SIAM International Conference on Data Mining (SDM16)**, May 5 - 7, 2016 at Hilton Miami Downtown, Miami, Florida, USA. SIAM has sole use for distribution in all forms and media, such as microfilm and anthologies, except that the author(s) or, in the case of a "work made for hire," the employer will retain:

The right to use all or part of the content of the paper in future works of the author(s), including author's teaching, technical collaborations, conference presentations, lectures, or other scholarly works and professional activities as well as to the extent the fair use provisions of the U.S. Copyright Act permit. If the copyright is granted to SIAM, then the proper notice of the SIAM's copyright should be provided.

The right to post the final draft of the paper on noncommercial pre-print serves like arXiv.org.

The right to post the final version of the paper on the author's personal web site and on the web server of the author's institution, provided the proper notice of the SIAM's copyright is included and that no separate or additional fees are collected for access to or distribution of the paper.

The right to refuse permission to third parties to republish all or part of the paper or translation thereof.

It is affirmed that neither this paper nor portions of it have been published elsewhere and that a copyright transfer agreement has not been signed permitting the publication of a similar paper in a journal or elsewhere. For multi-author works, the signing author agrees to notify all co-authors of his/her action.

Transfer of Copyright to the Publisher

SIAM strongly recommends this option. This transfer of copyright provides SIAM the legal basis not only to publish and to distribute the work, but also to pursue infringements of copyright (such as plagiarism and other forms of unauthorized use) and to grant permissions for the legitimate uses of the work by third parties. This option should not be selected if the work was prepared by a government office or employee as part of his or her official duties.

[__] By selecting the box at left, the Author hereby irrevocably assigns, conveys and transfers the copyright to the Work to SIAM. SIAM shall have sole rights of
distribution and/or publication of the work in all forms and media, throughout the world, except for those rights given to the Author above.

[__] By selecting the box at left, the Author DOES NOT assign, convey and transfer the

.2 ECML-PKDD Copyright Letter

Consent to Publish

Lecture Notes in Computer Science

Title of the Book or Conference Name: . . . . . .

Volume Editor(s) Name(s): . . . . . .

Title of the Contribution: . . . . . .

Author(s) Full Name(s): . . . . . .

Corresponding Author's Name, Affiliation Address, and Email:

. . . . . .

. . . . . .

When Author is more than one person the expression "Author" as used in this agreement will apply collectively unless otherwise indicated.

The Publisher intends to publish the Work under the imprint **Springer**. The Work may be published in the book series **Lecture Notes in Computer Science (LNCS, LNAI or LNBI)**.

§ 1 Rights Granted

Author hereby grants and assigns to **Springer Nature Switzerland AG, Gewerbestrasse 11, 6330 Cham, Switzerland** (hereinafter called **Publisher**) the exclusive, sole, permanent, world-wide, transferable, sub-licensable and unlimited right to reproduce, publish, distribute, transmit, make available or otherwise communicate to the public, translate, publicly perform, archive, store, lease or lend and sell the Contribution or parts thereof individually or together with other works in any language, in all revisions and versions (including soft cover, book club and collected editions, anthologies, advance printing, reprints or print to order, microfilm editions, audiograms and videograms), in all forms and media of expression including in electronic form (including offline and online use, push or pull technologies, use in databases and data networks (e.g. the Internet) for display, print and storing on any and all stationary or portable end-user devices, e.g. text readers, audio, video or interactive devices, and for use in multimedia or interactive versions as well as for the display or transmission of the Contribution or parts thereof in data networks or search engines, and posting the Contribution on social media accounts closely related to the Work), in whole, in part or in abridged form, in each case as now known or developed in the future, including the right to grant further time-limited or permanent rights. Publisher especially has the right to permit others to use individual illustrations, tables or text quotations and may use the Contribution for advertising purposes. For the purposes of use in electronic forms, Publisher may adjust the Contribution to the respective form of use and include links (e.g. frames or inline-links) or otherwise combine it with other works and/or remove links or combinations with other works provided in the Contribution. For the avoidance of doubt, all provisions of this contract apply regardless of whether the Contribution and/or the Work itself constitutes a database under applicable copyright laws or not.

The copyright in the Contribution shall be vested in the name of Publisher. Author has asserted his/her right(s) to be identified as the originator of this Contribution in all editions and versions of the Work and parts thereof, published in all forms and media. Publisher may take, either in its own name or in that of Author, any necessary steps to protect the rights granted under this Agreement against infringement by third parties. It will have a copyright notice inserted into all editions of the Work according to the provisions of the Universal Copyright Convention (UCC).

The parties acknowledge that there may be no basis for claim of copyright in the United States to a Contribution prepared by an officer or employee of the United States government as part of that person's official duties. If the Contribution was performed under a United States government contract, but Author is not a United States government employee, Publisher grants the United States government royalty-free permission to reproduce all or part of the Contribution and to authorise others to do so for United States government purposes. If the Contribution was prepared or published by or under the direction or control of the Crown (i.e., the constitutional monarch of the Commonwealth realm) or any Crown government department, the copyright in the Contribution shall, subject to any agreement with Author, belong to the Crown. If Author is an officer or employee of the United States government or of the Crown, reference will be made to this status on the signature page.

§ 2 Rights Retained by Author

Author retains, in addition to uses permitted by law, the right to communicate the content of the Contribution to other research colleagues, to share the Contribution with them in manuscript form, to perform or present the Contribution or to use the content for non-commercial internal and educational purposes, provided the original source of publication is cited according to the current citation standards in any printed or electronic materials. Author retains the right to republish the Contribution in any collection consisting solely of Author's own works without charge, subject to ensuring that the publication of the Publisher is properly credited and that the relevant copyright notice is repeated verbatim. Author may self-archive an author-created version of his/her Contribution on his/her own website and/or the repository of Author's department or faculty. Author may also deposit this version on his/her funder's or funder's designated repository at the funder's request or as a result of a legal obligation. He/she may not use the Publisher's PDF version, which is posted on the Publisher's platforms, for the purpose of self-archiving or deposit. Furthermore, Author may only post his/her own version, provided acknowledgment is given to the original source of publication and a link is inserted to the published article on the Publisher's website. The link must be provided by inserting the DOI number of the article in the following sentence: "The final authenticated version is available online at https://doi.org/[insert DOI]." The DOI (Digital Object Identifier) can be found at the bottom of the first page of the published paper.

Prior versions of the Contribution published on non-commercial pre-print servers like ArXiv/CoRR and HAL can remain on these servers and/or can be updated with Author's accepted version. The final published version (in pdf or html/xml format) cannot be used for this purpose. Acknowledgment needs to be given to the final publication and a link must be inserted to the published Contribution on the Publisher's website, by inserting the DOI number of the article in the following sentence: "The final authenticated publication is available online at https://doi.org/[insert DOI]".

Author retains the right to use his/her Contribution for his/her further scientific career by including the final published paper in his/her dissertation or doctoral thesis provided acknowledgment is given to the original source of publication. Author also retains the right to use, without having to pay a fee and without having to inform the Publisher, parts of the Contribution (e.g. illustrations) for inclusion in future work. Authors may publish an extended version of their proceedings paper as a journal article provided the following principles are adhered to: a) the extended version includes at least 30% new material, b) the original publication is cited, and c) it includes an explicit statement about the increment (e.g., new results, better description of materials, etc.).

§ 3 Warranties

Author agrees, at the request of Publisher, to execute all documents and do all things reasonably required by Publisher in order to confer to Publisher all rights intended to be granted under this Agreement. Author warrants that the Contribution is original except for such excerpts from copyrighted works (including illustrations, tables, animations and text quotations) as may be included with the permission of the copyright holder thereof, in which case(s) Author is required to obtain written permission to the extent necessary and to indicate the precise sources of the excerpts in the manuscript. Author is also requested to store the signed permission forms and to make them available to Publisher if required.

Author warrants that Author is entitled to grant the rights in accordance with Clause 1 "Rights Granted", that Author has not assigned such rights to third parties, that the Contribution has not heretofore been published in whole or in part, that the Contribution contains no libellous or defamatory statements and does not infringe on any copyright, trademark, patent, statutory right or proprietary right of others, including rights obtained through licences; and that Author will indemnify Publisher against any costs, expenses or damages for which Publisher may become liable as a result of any claim which, if true, would constitute a breach by Author of any of Author's representations or warranties in this Agreement.

Author agrees to amend the Contribution to remove any potential obscenity, defamation, libel, malicious falsehood or otherwise unlawful part(s) identified at any time. Any such removal or alteration shall not affect the warranty and indemnity given by Author in this Agreement.

§ 4 Delivery of Contribution and Publication

Author agrees to deliver to the responsible Volume Editor (for conferences, usually one of the Program Chairs), on a date to be agreed upon, the manuscript created according to the Publisher's Instructions for Authors. Publisher will undertake the reproduction and distribution of the Contribution at its own expense and risk. After submission of the Consent to Publish form signed by the Corresponding Author, changes of authorship, or in the order of the authors listed, will not be accepted by the Publisher.

.3 ACM Copyright Letter

"Authors can reuse any portion of their own work in a new work of their own (and no fee is expected) as long as a citation and DOI pointer to the Version of Record in the ACM Digital Library are included.

Contributing complete papers to any edited collection of reprints for which the author is not the editor, requires permission and usually a republication fee.

Authors can include partial or complete papers of their own (and no fee is expected) in a dissertation as long as citations and DOI pointers to the Versions of Record in the ACM Digital Library are included. Authors can use any portion of their own work in presentations and in the classroom (and no fee is expected)." [1]

---

[1] `http://authors.acm.org/main.html`

.4 IEEE Copyright Letter

Title: Multi-view Graph Embedding with Hub Detection for Brain Network Analysis

Conference Proceedings: 2017 IEEE International Conference on Data Mining (ICDM)

Author: Guixiang Ma

Publisher: IEEE

Date: Nov. 2017

Thesis / Dissertation Reuse

The IEEE does not require individuals working on a thesis to obtain a formal reuse license, however, you may print out this statement to be used as a permission grant:

Requirements to be followed when using any portion (e.g., figure, graph, table, or textual material) of an IEEE copyrighted paper in a thesis:

1) In the case of textual material (e.g., using short quotes or referring to the work within these papers) users must give full credit to the original source (author, paper, publication) followed by the IEEE copyright line © 2011 IEEE.
2) In the case of illustrations or tabular material, we require that the copyright line © [Year of original publication] IEEE appear prominently with each reprinted figure and/or table.
3) If a substantial portion of the original paper is to be used, and if you are not the senior author, also obtain the senior author's approval.

Requirements to be followed when using an entire IEEE copyrighted paper in a thesis:

1) The following IEEE copyright/ credit notice should be placed prominently in the references: © [year of original publication] IEEE. Reprinted, with permission, from [author names, paper title, IEEE publication title, and month/year of publication]
2) Only the accepted version of an IEEE copyrighted paper can be used when posting the paper or your thesis on-line.
3) In placing the thesis on the author's university website, please display the following message in a prominent place on the website: In reference to IEEE copyrighted material which is used with permission in this thesis, the IEEE does not endorse any of [university/educational entity's name goes here]'s products or services. Internal or personal use of this material is permitted. If interested in reprinting/republishing IEEE copyrighted material for advertising or promotional purposes or for creating new collective works for resale or redistribution, please go to http://www.ieee.org/publications_standards/publications/rights/rights_link.html to learn how to obtain a License from RightsLink.

If applicable, University Microfilms and/or ProQuest Library, or the Archives of Canada may supply single copies of the dissertation.

CITED LITERATURE

[Absil et al. , 2009]Absil, P.-A., Mahony, R., and Sepulchre, R.: Optimization algorithms on matrix manifolds. Princeton University Press, 2009.

[Aggarwal et al. , 2007]Aggarwal, C. C., Ta, N., Wang, J., Feng, J., and Zaki, M.: Xproj: a framework for projected structural clustering of xml documents. In KDD, pages 46–55. ACM, 2007.

[Aggarwal and Wang, 2010]Aggarwal, C. C. and Wang, H.: A survey of clustering algorithms for graph data. In Managing and mining graph data, pages 275–301. Springer, 2010.

[Balcan et al. , 2006]Balcan, M.-F., Blum, A., and Vempala, S.: Kernels as features: On kernels, margins, and low-dimensional mappings. Machine Learning, 65(1):79–94, 2006.

[Barnathan et al. , 2010]Barnathan, M., Megalooikonomou, V., Faloutsos, C., Mohamed, F. B., and Faro, S.: Twave: High-order analysis of spatiotemporal data. In Pacific-Asia Conference on Knowledge Discovery and Data Mining, pages 246–253. Springer, 2010.

[Belkin and Niyogi, 2001]Belkin, M. and Niyogi, P.: Laplacian eigenmaps and spectral techniques for embedding and clustering. In NIPS, 2001.

[Berkhin, 2006]Berkhin, P.: A survey of clustering data mining techniques. In Grouping multidimensional data, pages 25–71. Springer, 2006.

[Berlinet and Thomas-Agnan, 2011]Berlinet, A. and Thomas-Agnan, C.: Reproducing kernel Hilbert spaces in probability and statistics. Springer Science & Business Media, 2011.

[Bickel and Scheffer, 2004]Bickel, S. and Scheffer, T.: Multi-view clustering. In ICDM, 2004.

[Boyd et al. , 2011]Boyd, S., Parikh, N., Chu, E., Peleato, B., and Eckstein, J.: Distributed optimization and statistical learning via the alternating direction method of multipliers. Foundations and Trends® in Machine Learning, 3(1):1–122, 2011.

[Brandes, 2001]Brandes, U.: A faster algorithm for betweenness centrality. Journal of mathematical sociology, 25(2):163–177, 2001.

[Cahill et al. , 2016]Cahill, N. D., Singh, H., Zhang, C., Corcoran, D. A., Prengaman, A. M., Wenger, P. S., Hamilton, J. F., Bajorski, P., and Michael, A. M.: Multiple-view spectral clustering for group-wise functional community detection. arXiv preprint arXiv:1611.06981, 2016.

[Cai, 2011]Cai, D.: Litekmeans: the fastest matlab implementation of kmeans. Software available at: http://www. zjucadcg. cn/dengcai/Data/Clustering. html, 2011.

[Cai et al. , 2013]Cai, X., Nie, F., Cai, W., and Huang, H.: Heterogeneous image features integration via multi-modal semi-supervised learning model. In Proceedings of the IEEE International Conference on Computer Vision, pages 1737–1744, 2013.

[Cai et al. , 2011]Cai, X., Nie, F., Huang, H., and Kamangar, F.: Heterogeneous image feature integration via multi-modal spectral clustering. In Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on, pages 1977–1984. IEEE, 2011.

[Cao et al. , 2017]Cao, B., He, L., Wei, X., Xing, M., Yu, P. S., Klumpp, H., and Leow, A. D.: t-BNE: Tensor-based brain network embedding. In Proceedings of SIAM International Conference on Data Mining (SDM), 2017.

[Cao et al. , 2015a]Cao, B., Kong, X., Zhang, J., Philip, S. Y., and Ragin, A. B.: Identifying hiv-induced subgraph patterns in brain networks with side information. Brain informatics, 2(4):211–223, 2015.

[Cao et al. , 2015b]Cao, B., Kong, X., Zhang, J., Philip, S. Y., and Ragin, A. B.: Mining brain networks using multiple side views for neurological disorder identification. In ICDM, pages 709–714. IEEE, 2015.

[Cao et al. , 2015c]Cao, B., Zhan, L., Kong, X., Yu, P. S., Vizueta, N., Altshuler, L. L., and Leow, A. D.: Identification of discriminative subgraph patterns in fmri brain networks in bipolar affective disorder. In International Conference on Brain Informatics and Health, pages 105–114. Springer, 2015.

[Chang and Lin, 2011]Chang, C.-C. and Lin, C.-J.: Libsvm: a library for support vector machines. ACM transactions on intelligent systems and technology (TIST), 2(3):27, 2011.

[Chaudhuri et al. , 2009]Chaudhuri, K., Kakade, S. M., Livescu, K., and Sridharan, K.: Multi-view clustering via canonical correlation analysis. In ICML, 2009.

[Cilla Ugarte, 2012]Cilla Ugarte, R.: Action recognition in visual sensor networks: a data fusion perspective. 2012.

[Craddock et al. , 2012]Craddock, R. C., James, G. A., Holtzheimer, P. E., Hu, X. P., and Mayberg, H. S.: A whole brain fmri atlas generated via spatially constrained spectral clustering. Human brain mapping, 33(8):1914–1928, 2012.

[Cressie and Wikle, 2015]Cressie, N. and Wikle, C. K.: Statistics for spatio-temporal data. John Wiley & Sons, 2015.

[Cristianini et al. , 2000]Cristianini, N., Shawe-Taylor, J., et al.: An introduction to support vector machines and other kernel-based learning methods. Cambridge university press, 2000.

[Crossley et al. , 2014]Crossley, N. A., Mechelli, A., Scott, J., Carletti, F., Fox, P. T., McGuire, P., and Bullmore, E. T.: The hubs of the human connectome are generally implicated in the anatomy of brain disorders. Brain, 137(8):2382–2395, 2014.

[De Graaf and Koch, 2011]De Graaf, R. and Koch, K.: Methods and apparatus for compensating field inhomogeneities in magnetic resonance studies, October 11 2011. US Patent 8,035,387.

[De Silva and Lim, 2008]De Silva, V. and Lim, L.-H.: Tensor rank and the ill-posedness of the best low-rank approximation problem. SIAM Journal on Matrix Analysis and Applications, 30(3):1084–1127, 2008.

[Deng et al. , 2013]Deng, F., Zhu, D., Lv, J., Guo, L., and Liu, T.: Fmri signal analysis using empirical mean curve decomposition. IEEE transactions on biomedical engineering, 60(1):42–54, 2013.

[Donath and Hoffman, 1973]Donath, W. E. and Hoffman, A. J.: Lower bounds for the partitioning of graphs. IBM Journal of Research and Development, 17(5):420–425, 1973.

[Ecker et al. , 2010]Ecker, C., Rocha-Rego, V., Johnston, P., Mourao-Miranda, J., Marquand, A., Daly, E. M., Brammer, M. J., Murphy, C., Murphy, D. G., Consortium, M. A., et al.: Investigating the predictive value of whole-brain structural mr scans in autism: a pattern classification approach. Neuroimage, 49(1):44–56, 2010.

[Fang and Zhang, 2013]Fang, Z. and Zhang, Z. M.: Discriminative feature selection for multi-view cross-domain learning. In CIKM, pages 1321–1330. ACM, 2013.

[Fink and Gandhi, 2011]Fink, E. and Gandhi, H. S.: Compression of time series by extracting major extrema. Journal of Experimental & Theoretical Artificial Intelligence, 23(2):255–270, 2011.

[Frey and Dueck, 2007]Frey, B. J. and Dueck, D.: Clustering by passing messages between data points. Science, 315(5814):972–976, 2007.

[Fu and Ma, 2012]Fu, Y. and Ma, Y.: Graph embedding for pattern analysis. Springer Science & Business Media, 2012.

[Gao et al. , 2013]Gao, J., Du, N., Fan, W., Turaga, D., Parthasarathy, S., and Han, J.: A multi-graph spectral framework for mining multi-source anomalies. In Graph Embedding for Pattern Analysis, pages 205–227. Springer, 2013.

[Gärtner, 2003]Gärtner, T.: A survey of kernels for structured data. ACM SIGKDD Explorations Newsletter, 5(1):49–58, 2003.

[Greene and Cunningham, 2009]Greene, D. and Cunningham, P.: A matrix factorization approach for integrating multiple data views. In ECML/PKDD, pages 423–438. Springer, 2009.

[Haller et al. , 2007]Haller, S., Klarhoefer, M., Schwarzbach, J., Radue, E. W., and Indefrey, P.: Spatial and temporal analysis of fmri data on word and sentence reading. European Journal of Neuroscience, 26(7):2074–2084, 2007.

[Hao et al. , 2013]Hao, Z., He, L., Chen, B., and Yang, X.: A linear support higher-order tensor machine for classification. IEEE Transactions on Image Processing, 22(7):2911–2920, 2013.

[Harshman et al. , 2003]Harshman, R. A., Hong, S., and Lundy, M. E.: Shifted factor analysispart i: Models and properties. Journal of Chemometrics: A Journal of the Chemometrics Society, 17(7):363–378, 2003.

[He et al. , 2014a]He, L., Kong, X., Yu, P. S., Yang, X., Ragin, A. B., and Hao, Z.: Dusk: A dual structure-preserving kernel for supervised tensor learning with applications to neuroimages. In Proceedings of the 2014 SIAM International Conference on Data Mining, pages 127–135. SIAM, 2014.

[He et al. , 2014b]He, L., Kong, X., Yu, P. S., Yang, X., Ragin, A. B., and Hao, Z.: Dusk: A dual structure-preserving kernel for supervised tensor learning with applications to neuroimages. In SDM, 2014.

[He et al. , 2017]He, L., Lu, C.-T., Ma, G., Wang, S., Shen, L., Philip, S. Y., and Ragin, A. B.: Kernelized support tensor machines. In International Conference on Machine Learning, pages 1442–1451, 2017.

[He et al. , 2016a]He, L., Lu, C.-T., Ma, J., Cao, J., Shen, L., and Philip, S. Y.: Joint community and structural hole spanner detection via harmonic modularity. In KDD. ACM, 2016.

[He et al. , 2016b]He, L., Lu, C.-T., Ma, J., Cao, J., Shen, L., and Yu, P. S.: Joint community and structural hole spanner detection via harmonic modularity. In Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 875–884. ACM, 2016.

[He et al. , 2012a]He, R., Tan, T., Wang, L., and Zheng, W.-S.: $\ell_{2,1}$ regularized correntropy for robust feature selection. In CVPR, pages 2504–2511, 2012.

[He et al. , 2012b]He, R., Tan, T., Wang, L., and Zheng, W.-S.: l 2, 1 regularized correntropy for robust feature selection. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 2504–2511. IEEE, 2012.

[Huang et al. , 2012]Huang, H.-C., Chuang, Y.-Y., and Chen, C.-S.: Affinity aggregation for spectral clustering. In Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on, pages 773–780. IEEE, 2012.

[Jie et al. , 2014]Jie, B., Zhang, D., Gao, W., Wang, Q., Wee, C., and Shen, D.: Integration of network topological and connectivity properties for neuroimaging classification. Biomedical Engineering, 61(2):576, 2014.

[Jordanov and Jain, 2010]Jordanov, I. and Jain, R.: Knowledge-Based and Intelligent Information and Engineering Systems. Springer, 2010.

[Kaiser, 2011]Kaiser, M.: A tutorial in connectome analysis: topological and spatial features of brain networks. Neuroimage, 57(3):892–907, 2011.

[Kalayeh et al. , 2014]Kalayeh, M. M., Idrees, H., and Shah, M.: Nmf-knn: Image annotation using weighted multi-view non-negative matrix factorization. In CVPR, 2014.

[King, 2002]King, R. B.: Topological methods in chemical structure and bonding. Encyclopedia of Computational Chemistry, 5, 2002.

[Koch et al. , 2012]Koch, W., Teipel, S., Mueller, S., Benninghoff, J., Wagner, M., Bokde, A. L., Hampel, H., Coates, U., Reiser, M., and Meindl, T.: Diagnostic power of default mode network resting state fmri in the detection of alzheimer's disease. Neurobiology of aging, 33(3):466–478, 2012.

[Kolda and Bader, 2009]Kolda, T. G. and Bader, B. W.: Tensor decompositions and applications. SIAM review, 51(3):455–500, 2009.

[Kong et al. , 2013]Kong, X., Ragin, A. B., Wang, X., and Yu, P. S.: Discriminative feature selection for uncertain graph classification. In SDM, pages 82–93. SIAM, 2013.

[Kong and Yu, 2014]Kong, X. and Yu, P. S.: Brain network analysis: a data mining perspective. ACM SIGKDD Explorations Newsletter, 15(2):30–38, 2014.

[Kuang et al. , 2013]Kuang, L.-D., Lin, Q.-H., Gong, X.-F., Fan, J., Cong, F.-Y., and Calhoun, V. D.: Multi-subject fmri data analysis: Shift-invariant tensor factorization vs. group independent component analysis. In Signal and Information Processing (ChinaSIP), 2013 IEEE China Summit & International Conference on, pages 269–272. IEEE, 2013.

[Kuhn, 1955]Kuhn, H. W.: The hungarian method for the assignment problem. Naval research logistics quarterly, 2(1-2):83–97, 1955.

[Kumar et al. , 2011]Kumar, A., Rai, P., and Daume, H.: Co-regularized multi-view spectral clustering. In NIPS, 2011.

[Kuo et al. , 2015]Kuo, C.-T., Wang, X., Walker, P., Carmichael, O., Ye, J., and Davidson, I.: Unified and contrasting cuts in multiple graphs: application to medical imaging segmentation. In KDD, pages 617–626. ACM, 2015.

[Leow et al. , 2013]Leow, A., Ajilore, O., Zhan, L., Arienzo, D., GadElkarim, J., Zhang, A., Moody, T., Van Horn, J., Feusner, J., Kumar, A., et al.: Impaired inter-hemispheric integration in bipolar disorder revealed with brain network analyses. Biological psychiatry, 73(2):183–193, 2013.

[Li et al. , 2015]Li, Y., Nie, F., Huang, H., and Huang, J.: Large-scale multi-view spectral clustering via bipartite graph. In AAAI, pages 2750–2756, 2015.

[Lian et al. , 2004]Lian, W., Mamoulis, N., Yiu, S.-M., et al.: An efficient and scalable algorithm for clustering xml documents by structure. IEEE transactions on Knowledge and Data Engineering, 16(1):82–96, 2004.

[Lindquist and others, 2008]Lindquist, M. A. et al.: The statistical analysis of fmri data. Statistical science, 23(4):439–464, 2008.

[Liu et al. , 2013]Liu, J., Wang, C., Gao, J., and Han, J.: Multi-view clustering via joint nonnegative matrix factorization. In SDM, 2013.

[Lu et al. , 2017]Lu, C.-T., He, L., Shao, W., Cao, B., and Yu, P. S.: Multilinear factorization machines for multi-task multi-view learning. In Proceedings of the Tenth ACM International Conference on Web Search and Data Mining, pages 701–709. ACM, 2017.

[Ma et al. , 2016]Ma, G., He, L., Cao, B., Zhang, J., Philip, S. Y., and Ragin, A. B.: Multi-graph clustering based on interior-node topology with applications to brain networks. In Joint European Conference on Machine Learning and Knowledge Discovery in Databases, pages 476–492. Springer, 2016.

[Ma et al. , 2017]Ma, G., He, L., Lu, C.-T., Shao, W., Yu, P. S., Leow, A. D., and Ragin, A. B.: Multi-view clustering with graph embedding for connectome analysis. In Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, pages 127–136. ACM, 2017.

[Ma et al. , 2016]Ma, G., He, L., Lu, C.-T., Yu, P. S., Shen, L., and Ragin, A. B.: Spatio-temporal tensor analysis for whole-brain fmri classification. In Proceedings of the 2016 SIAM International Conference on Data Mining, pages 819–827. SIAM, 2016.

[Ma et al. , 2017]Ma, G., Lu, C.-T., He, L., Philip, S. Y., and Ann, B. R.: Multi-view graph embedding with hub detection for brain network analysis. In ICDM, 2017.

[Matthews et al. , 2006]Matthews, P. M., Honey, G. D., and Bullmore, E. T.: Neuroimaging: Applications of fmri in translational medicine and clinical practice. Nature Reviews Neuroscience, 7(9):732, 2006.

[McKeown et al. , 2007]McKeown, M. J., Li, J., Huang, X., Lewis, M. M., Rhee, S., Truong, K. Y., and Wang, Z. J.: Local linear discriminant analysis (llda) for group and region of interest (roi)-based fmri analysis. Neuroimage, 37(3):855–865, 2007.

[Mørup et al. , 2008]Mørup, M., Hansen, L. K., Arnfred, S. M., Lim, L.-H., and Madsen, K. H.: Shift-invariant multilinear decomposition of neuroimaging data. NeuroImage, 42(4):1439–1450, 2008.

[Mousazadeh and Cohen, 2015]Mousazadeh, S. and Cohen, I.: Embedding and function extension on directed graph. Signal Processing, 111:137–149, 2015.

[Nie et al. , 2016]Nie, F., Li, J., Li, X., et al.: Parameter-free auto-weighted multiple graph learning: A framework for multiview clustering and semi-supervised classification. International Joint Conferences on Artificial Intelligence, 2016.

[Nie et al. , 2011]Nie, F., Zeng, Z., Tsang, I. W., Xu, D., and Zhang, C.: Spectral embedded clustering: A framework for in-sample and out-of-sample spectral clustering. IEEE Trans on Neural Networks, 22(11):1796–1808, 2011.

[Nikolova and Ng, 2005]Nikolova, M. and Ng, M. K.: Analysis of half-quadratic minimization methods for signal and image recovery. SIAM Journal on Scientific computing, 27(3):937–966, 2005.

[Oikonomou et al. , 2012]Oikonomou, V. P., Blekas, K., and Astrakas, L.: A sparse and spatially constrained generative regression model for fmri data analysis. IEEE Transactions on Biomedical Engineering, 59(1):58–67, 2012.

[Onnela et al. , 2005]Onnela, J.-P., Saramäki, J., Kertész, J., and Kaski, K.: Intensity and coherence of motifs in weighted complex networks. Physical Review E, 71(6):065103, 2005.

[Ou et al. , 2016]Ou, M., Cui, P., Pei, J., Zhang, Z., and Zhu, W.: Asymmetric transitivity preserving graph embedding. In SIGKDD, 2016.

[Papalexakis et al. , 2013]Papalexakis, E. E., Akoglu, L., and Ience, D.: Do more views of a graph help? community detection and clustering in multi-graphs. In Information fusion (FUSION), 2013 16th international conference on, pages 899–905. IEEE, 2013.

[Ragin et al. , 2012a]Ragin, A. B., Du, H., Ochs, R., Wu, Y., Sammet, C. L., Shoukry, A., and Epstein, L. G.: Structural brain alterations can be detected early in hiv infection. Neurology, 79(24):2328–2334, 2012.

[Ragin et al. , 2012b]Ragin, A. B., Du, H., Ochs, R., Wu, Y., Sammet, C. L., Shoukry, A., and Epstein, L. G.: Structural brain alterations can be detected early in hiv infection. Neurology, 79(24):2328–2334, 2012.

[Rezvani et al. , 2015]Rezvani, M., Liang, W., Xu, W., and Liu, C.: Identifying top-k structural hole spanners in large-scale social networks. In Proceedings of the 24th ACM International on Conference on Information and Knowledge Management, pages 263–272. ACM, 2015.

[Roweis and Saul, 2000]Roweis, S. T. and Saul, L. K.: Nonlinear dimensionality reduction by locally linear embedding. Science, 290(5500):2323–2326, 2000.

[Saul and Roweis, 2000]Saul, L. K. and Roweis, S. T.: An introduction to locally linear embedding. http://www. cs. toronto. edu/~ roweis/lle/publications. html, 2000.

[Schölkopf et al. , 2001]Schölkopf, B., Herbrich, R., and Smola, A. J.: A generalized representer theorem. In International conference on computational learning theory, pages 416–426. Springer, 2001.

[Shao et al. , 2016]Shao, W., He, L., Lu, C.-T., Wei, X., and Philip, S. Y.: Online unsupervised multi-view feature selection. In ICDM, 2016.

[Shao et al. , 2015]Shao, W., He, L., and Philip, S. Y.: Clustering on multi-source incomplete data via tensor modeling and factorization. In Pacific-Asia Conference on Knowledge Discovery and Data Mining, pages 485–497. Springer, 2015.

[Shi and Malik, 2000]Shi, J. and Malik, J.: Normalized cuts and image segmentation. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 22(8):888–905, 2000.

[Signoretto et al. , 2011]Signoretto, M., De Lathauwer, L., and Suykens, J. A.: A kernel-based framework to tensorial data analysis. Neural networks, 24(8):861–874, 2011.

[Song et al. , 2015]Song, X., Meng, L., Shi, Q., and Lu, H.: Learning tensor-based features for whole-brain fmri classification. In International Conference on Medical Image Computing and Computer-Assisted Intervention, pages 613–620. Springer, 2015.

[Spielman, 2010]Spielman, D. A.: Algorithms, graph theory, and linear equations in laplacian matrices. In ICM, volume 4, pages 2698–2722, 2010.

[Sporns et al. , 2005]Sporns, O., Tononi, G., and Kötter, R.: The human connectome: a structural description of the human brain. PLoS Comput Biol, 1(4):e42, 2005.

[Tzourio-Mazoyer et al. , 2002]Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., and Joliot, M.: Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. Neuroimage, 15(1):273–289, 2002.

[Van Den Heuvel and Sporns, 2011]Van Den Heuvel, M. P. and Sporns, O.: Rich-club organization of the human connectome. Journal of Neuroscience, 31(44):15775–15786, 2011.

[van den Heuvel and Sporns, 2013]van den Heuvel, M. P. and Sporns, O.: Network hubs in the human brain. Trends in cognitive sciences, 17(12):683–696, 2013.

[Vapnik, 2013]Vapnik, V.: The nature of statistical learning theory. Springer science & business media, 2013.

[Vishwanathan et al. , 2010]Vishwanathan, S. V. N., Schraudolph, N. N., Kondor, R., and Borgwardt, K. M.: Graph kernels. Journal of Machine Learning Research, 11(Apr):1201–1242, 2010.

[Von Luxburg, 2007]Von Luxburg, U.: A tutorial on spectral clustering. Statistics and computing, 17(4):395–416, 2007.

[Wang et al. , 2011a]Wang, X., Foryt, P., Ochs, R., Chung, J.-H., Wu, Y., Parrish, T., and Ragin, A. B.: Abnormalities in resting-state functional connectivity in early human immunodeficiency virus infection. Brain connectivity, 1(3):207–217, 2011.

[Wang et al. , 2011b]Wang, X., Foryt, P., Ochs, R., Chung, J.-H., Wu, Y., Parrish, T., and Ragin, A. B.: Abnormalities in resting-state functional connectivity in early human immunodeficiency virus infection. Brain connectivity, 1(3):207–217, 2011.

[Wang et al. , 2015]Wang, Y., Chen, R., Ghosh, J., Denny, J. C., Kho, A., Chen, Y., Malin, B. A., and Sun, J.: Rubik: Knowledge guided tensor factorization and completion for health data analytics. In SIGKDD, 2015.

[Wee et al. , 2012]Wee, C.-Y., Yap, P.-T., Zhang, D., Denny, K., Browndyke, J. N., Potter, G. G., Welsh-Bohmer, K. A., Wang, L., and Shen, D.: Identification of mci individuals using structural and functional connectivity networks. Neuroimage, 59(3):2045–2056, 2012.

[Wen and Yin, 2013]Wen, Z. and Yin, W.: A feasible method for optimization with orthogonality constraints. Mathematical Programming, 142(1-2):397–434, 2013.

[Whitfield-Gabrieli and Nieto-Castanon, 2012]Whitfield-Gabrieli, S. and Nieto-Castanon, A.: Conn: a functional connectivity toolbox for correlated and anticorrelated brain networks. Brain connectivity, 2(3):125–141, 2012.

[Xia et al. , 2013]Xia, M., Wang, J., and He, Y.: Brainnet viewer: a network visualization tool for human brain connectomics. PloS one, 8(7):e68910, 2013.

[Xia et al. , 2010]Xia, T., Tao, D., Mei, T., and Zhang, Y.: Multiview spectral embedding. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 40(6):1438–1446, 2010.

[Xie et al. , 2011]Xie, B., Mu, Y., Tao, D., and Huang, K.: m-sne: Multiview stochastic neighbor embedding. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics), 41(4):1088–1096, 2011.

[Xu et al. , 2013]Xu, C., Tao, D., and Xu, C.: A survey on multi-view learning. arXiv preprint arXiv:1304.5634, 2013.

[Yan et al. , 2007]Yan, S., Xu, D., Zhang, B., Zhang, H.-J., Yang, Q., and Lin, S.: Graph embedding and extensions: a general framework for dimensionality reduction. IEEE transactions on pattern analysis and machine intelligence, 29(1):40–51, 2007.

[Ye et al. , 2008]Ye, J., Chen, K., Wu, T., Li, J., Zhao, Z., Patel, R., Bae, M., Janardan, R., Liu, H., Alexander, G., et al.: Heterogeneous data fusion for alzheimer's disease study. In Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining, pages 1025–1033. ACM, 2008.

[Yin et al. , 2016]Yin, M., Xie, S., Guo, Y., et al.: Low-rank multi-view clustering in third-order tensor space. arXiv preprint arXiv:1608.08336, 2016.

[Zhang and Yu, 2015a]Zhang, J. and Yu, P. S.: Community detection for emerging networks. In SDM. SIAM, 2015.

[Zhang and Yu, 2015b]Zhang, J. and Yu, P. S.: Mutual clustering across multiple heterogeneous networks. In IEEE BigData Congress, 2015.

[Zhang et al. , 2016]Zhang, J., Cao, B., Xie, S., Lu, C.-T., Yu, P. S., and Ragin, A. B.: Identifying connectivity patterns for brain diseases via multi-side-view guided deep architectures. In Proceedings of the 2016 SIAM International Conference on Data Mining, pages 36–44. SIAM, 2016.

[Zhang et al. , 2015]Zhang, L., Zhang, Q., Zhang, L., Tao, D., Huang, X., and Du, B.: Ensemble manifold regularized sparse low-rank approximation for multiview feature embedding. Pattern Recognition, 48(10):3102–3112, 2015.

[Zhao et al. , 2013]Zhao, Q., Zhou, G., Adali, T., Zhang, L., and Cichocki, A.: Kernelization of tensor-based models for multiway data analysis: Processing of multidimensional structured data. IEEE Signal Processing Magazine, 30(4):137–148, 2013.

VITA

Name: Guixiang Ma

EDUCATION:

M.E. in Computer Science and Technology, Beijing Jiaotong University, 2013.

B.S. in Computer Science and Technology, Liaoning Normal University, 2010.

PUBLICATIONS:

- Guixiang Ma, Nesreen K. Ahmed, Ted Wilke, Dipanjan Sengupta, Michael W. Cole, Nick Turk-Browne, and Philip S. Yu. *Similarity Learning with Higher-Order Proximity for Brain Network Analysis.* arXiv: 1811.02662, 2018.
- Shuaijun Ge, Guixiang Ma, Sihong Xie, and Philip S. Yu. *Securing Behavior-based Opinion Spam Detection.* In IEEE International Conference on Big Data **(IEEE BigData)**, 2018.
- Guixiang Ma, Bokai Cao, Philip S. Yu, Ann Ragin. *Hypertension induces changes in Brain Network Organization.* In 26th Annual Meeting of International Society for Magnetic Resonance in Medicine **(ISMRM)**, Jun 2018, Paris, France.
- Ann B. Ragin, Can Wu, Guixiang Ma, Sameer A. Ansari, Michael Markl, Susanne Schnell. *Aortic flow and cerebral hemodynamics in age-related brain volume loss.* In 26th Annual Meeting of International Society for Magnetic Resonance in Medicine **(ISMRM)**, Jun 2018, Paris, France.

- Guixiang Ma, Chun-Ta Lu, Lifang He, Philip S. Yu, Ann Ragin. *Multi-view Graph Embedding with Hub Detection for Brain Network Analysis.* In IEEE International Conference on Data Mining **(ICDM)**, 2017.
- Guixiang Ma, Lifang He, Chun-Ta Lu, Weixiang Shao, Philip S Yu, Alex D Leow, and Ann B Ragin. *Multi-view clustering with graph embedding for connectome analysis.* In ACM International Conference on Information and Knowledge Management **(CIKM)**, 2017.
- Lifang He, Chun-Ta Lu, Guixiang Ma, Shen Wang, Linlin Shen, Philip S. Yu, and Ann B. Ragin. *Kernelized Support Tensor Machines.* In International Conference on Machine Learning **(ICML)**, 2017.
- Guixiang Ma, Lifang He, Bokai Cao, Jiawei Zhang, Philip S. Yu, and Ann B. Ragin. *Multi-graph Clustering Based on Interior-Node Topology with Applications to Brain Networks.* In Joint European Conference on Machine Learning and Knowledge Discovery in Databases **ECML/PKDD**, 2016.
- 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 SIAM International Conference on Data Mining **(SDM)**, 2016.
- Guixiang Ma, Lifang He, Chun-Ta Lu, Philip S. Yu, Linlin Shen and Ann B. Ragin. STTK: Spatio-Temporal Tensor Kernel Modeling for fMRI Brain Image Analysis. In ACM SIGKDD Workshop on Brain Science **(BrainKDD)**, 2015.

- Xu Liang, Youli Qu and Guixiang Ma. *Research on Contrastive Viewpoint Summarization for Opinionated Texts.* In Journal of Interconnection Networks. 2013 Sep;14(03):1360003.
- Xu Liang, Youli Qu and Guixiang Ma. *Research on Extension LexRank in Summarization for Opinionated Texts.* In Parallel and Distributed Computing, Applications and Technologies **(PDCAT)**, 2012.
- Guixiang Ma, Youli Qu. *A local LDA based method for Latent Aspect Rating Analysis on reviews.* In IEEE International Conference on Signal Processing **(ICSP)**, 2012.