Inferring Macroscopic Brain Connectomes via Group-Sparse Factorization

by

Farzane Aminmansour

A thesis submitted in partial fulfillment of the requirements for the degree of

Master of Science

Department of Computing Science

University of Alberta

Abstract

Mapping the macrostructural connectivity of the living human brain is one of the primary goals of neuroscientists who study connectomics. The reconstruction of a brain's structural connectivity, aka its connectome, typically involves applying expert analysis to diffusion-weighted magnetic resonance imaging (dMRI). A data-driven approach – inferring the underlying model from data – could overcome the limitations of such human-based approaches and improve precision mappings for a novel brain. In this work, we explore a framework that facilitates applying learning algorithms to automatically extract brain connectomes. Using a tensor encoding to unify the representation of brain structure and diffusion information, we design a constrained objective function with a group-regularizer that prefers a biologically plausible structure for each bundle of neuronal axons, called a fascicle. We show that the objective is convex and has a unique solution, ensuring identifiable connectomes for an individual brain. We develop an efficient optimization strategy for this extremely high-dimensional sparse problem, by reducing the number of parameters using a greedy algorithm, called GreedyOrientation, designed specifically for the problem. We show that GreedyOrientation significantly improves on a standard greedy algorithm, called Orthogonal Matching Pursuit. We confirm that our method works effectively by reconstructing structural connectivity of two major tracts. We conclude with an analysis of the solutions found by our method, showing it can accurately reconstruct the diffusion information while maintaining contiguous fascicles with smooth direction changes.

Preface

The major part of the contribution in this thesis have been published as a conference paper in Advances in Neural Information Processing Systems (NeurIPS) 2019 [5]. We are going to submit the results of new experiments as an extended version paper to a journal.

Acknowledgements

Contents

List of Figures

Glossary

Anisotropic diffusion signal The directional diffusion signal of the water molecule in the presence of a restricted barrier. 4

Atoms The basic elements to build the dictionary. 28

Connectome The structure of white matter connectivity in the human brain. 1

Constrained Spherical Deconvolution (CSD) An estimation of a white matter fibre Orientation Distribution Function (fODF) based on an estimate of the signal expected for a single-fibre white matter population (the so-called response function). 1

Diffusion Tensor Imaging (DTI) An MRI-based neuroimaging technique which makes it possible to estimate the location, orientation, and anisotropy of the brain's white matter tracts. 2

Diffusion-weighted magnetic resonance imaging (DWI or DW-MRI) The use of specific MRI sequences as well as software that generates images from the resulting data that uses the diffusion of water molecules to generate contrast in MR images. 4

Direction The gradient direction of the magnetic field. 18

Fascicle A bundle of neuronal axons wrapped with myelin sheaths. 1

Isotropic diffusion signal The non-directional diffusion signals of the water molecule in the absence of any restricted barrier. 4

MRI Magnetic resonance imaging is a medical imaging technique used in radiology to form pictures of the anatomy and the physiological processes of the body. MRI scanners use strong magnetic fields, magnetic field gradients, and radio waves to generate images of the organs in the body. 3

Node Each segment of a fascicle, which is straight enough to be represented with a diffusion ellipsoid can be considered as a node in a graph. Then, the whole fascicle could be built up by connecting those nodes to each other sequentially. In this thesis, nodes, orientations, and atoms are synonyms. 70

Orientation The orientation of longitudinal diffusion ellipsoid in the spatial space. 17

Tract The nerve fibres organized in bundles in the central nervous system. 2

Tractography A 3D modeling technique used to visually represent nerve tracts using data collected by diffusion MRI. 1

White matter The paler tissue of the brain and spinal cord, consisting mainly of nerve fibers with their myelin sheaths. 1

Chapter 1

Introduction

A fundamental challenge in neuroscience is to estimate the structure of White matter connectivity in the human brain, i.e., its Connectome [43, 101]. A connectome is made up of neuronal axon bundles wrapped with myelin sheaths, called Fascicle, that connect different areas of the brain. Acquiring information about the connectome is possible by measuring the diffusion of water molecules at different gradient directions of the magnetic field. Then, the fascicle structures can be inferred from the mathematical models of this diffusion-weighted signal by employing Tractography algorithms. In this chapter, we first explain variety of applications for estimating the the white matter structure. Then, we discuss the mathematical diffusion models and tractography methods, which are the current approaches to estimate the connectivity structure of the brain. Next, we elaborate the problem precisely and draw the scope of this project by introducing the input and output data. Finally, we describe our contributions in this project.

1.1   Applications

Currently, diffusion-weighted magnetic resonance imaging (dMRI) combined with fiber tractography is the only method available to extract structural brain connectomes in living human brains [11, 83, 102]. This method has revolutionized our understanding of the network structure of the human brain and the role of white matter in health and disease. In fact, different diffusion neuroimaging techniques, such as constrained spherical deconvolution (Con-

1

strained Spherical Deconvolution (CSD)) or diffusion tensor imaging (Diffusion Tensor Imaging (DTI)) as well as fiber tractography, may prove useful in clinical practice [42].

One application is in studying **normal brain development and aging** procedure. There are now many 2D and 3D atlases of the white matter architecture that could be generated from the quantitative diffusion data of a normal human brain by applying a tractography model [84, 87]. If done on the same individual at different ages, it will give us some insight about maturational changes of normal brains throughout his/her childhood.

Congenital anomalies, such as leukodystrophies, is a progressive disorder that is caused by a specific gene and mostly affects the spinal cord and/or peripheral nerves by destroying the myelin sheaths of the axons in the white matter. Applying neuroimaging techniques have been useful in evaluating such white matter Tract abnormalities [44].

Multiple Sclerosis (MS) is the most frequent neurodegenerative disease in young adults among **demyelinating and neurodegenerative diseases**, which is associated with destruction of myelin sheaths as well as axonal injuries. Characterizing the pathologic changes and integrity of white matter is critical for an effective treatment; this relies on having a sensitive and reliable in vivo method for estimating the white matter structure of the human brain.

Many research projects [45, 50, 51, 70] have shown that the number of fibers in a standardized region of interest (ROI) decreases in both **Normal Pressure Hydrocephalus (NPH) and Alzheimer disease**. Fiber tractography along with diffusion neuroimaging techniques can be useful in determining the severity of neurologic impairment and the response to therapy.

Identifying tumors and preoperative planning is a crucial step in every surgery [100]. Differentiating between the tumor and nontumor tissues as well as determining the tumor biology is a key step in identifying an appropriate therapy or surgery. Additionally, the extent of the disease must be determined prior to any therapy. Accordingly, differentiating between the tumor and necrosis tissues is also very important in surgery, chemotherapy, or radiation. All these evaluations are possible by utilizing neuroimaging techniques.

It is also beneficial to follow up the progress of a therapy by representing changes in a post-treatment MRI scan.

In the domain of **epilepsy and seizures**, elucidating the etiology of seizures and the underlying pathophysiology in chronic epilepsy could be investigated by characterizing microstructural abnormalities in epileptic foci. Neuroimaging techniques, such as dMRI and Positron Emission Tomography (PET)[1], make it possible to demonstrate the white matter fibers and tracts participating in the epileptic network [79].

About **ischemia and stroke**, many (including [63]), have shown that fiber tractography of DTI data can be used to localize stroke lesions in relation to functionally important pathways. Moreover, for assessing the crushed nerve fibers, where the part of the axon distal to the injury degenerates, DTI fiber tractography allows more accurate prognosis of long-term recovery or disability [113]. These methods are also useful in evaluating the ischemic injury to the developing brain of newborns and infants as well as elucidating alterations in brain connectivity resulting from neuroplasticity after stroke.

Another application is in **psychiatric disorders, dementia, and depression**. It consists of evaluating the structural differences of white matter architecture in psychiatric population and healthy controls, which is possible via DTI-tractography of the human brain. Torgerson et al. [95] have shown that the length and density of fiber tracts in healthy people are different from depressed or bipolar patients.

In **functional connectivity mapping and cognitive neuroscience**, understanding the pathways associated with a range of specialized cognitive functions in vivo, such as attention, perception, and decision making, is possible through probabilistic tractography methods to map the structural connectivity of the human brain [86].

Abnormal spinal cord diffusion signal as well as its fiber tractography provide physiologic information about disorganization of white matter for **spinal cord evaluation**. Furthermore, various DTI metrics are characterized to provide promises in diagnosing a range of spinal cord disorders ranging from

---

[1]A nuclear based medicine procedure to measure the metabolic activity of living tissues.

tumors, demyelination, and cord narrowing [53].

1.2 Mathematical Modeling of Diffusion Magnetic Resonance Imaging Signals

Measuring diffusion magnetic resonance imaging (dMRI) signals is a non-invasive method that reveals the tissue architecture by capturing the diffusion of non-free water molecules interacting with different obstacles, e.g. water molecule diffusion patterns in different brain tissues such as fibers or membranes [7]. A specific *sequence of MRI* scans of a single region (e.g. the head) of a person, then applying some software to generate the contrast in MR images from the diffusion data produces *Diffusion-Weighted Magnetic Resonance Imaging* (Diffusion-weighted magnetic resonance imaging (DWI or DW-MRI) or DW-MRI). Acquiring different MRI scan sequences of a person is possible by applying various setting of pulse sequences and pulsed field gradients, resulting in a particular image appearance.

In the MR images of the DWI data, the intensity of each volume pixel, known as a voxel, reflects the estimation of water diffusion rate at that location. This dMRI signal can be modeled as a linear combination of two components for each voxel. The first component reflects the directional diffusion signal, referred to as *Anisotropic diffusion signal diffusion*, which presumably related to the non-free water molecules inside the fibrous structure of axons covered with myelin sheaths [106]. The anisotropic diffusion is due to the presence of any constrained obstacle, meaning that water will diffuse more rapidly in the direction of internal structure rather than any other perpendicular directions (Figure 1.1). The second component is related to the non-directional signals, known as *Isotropic diffusion signal diffusion*, presumably originated from the diffusion of free water molecules in other brain tissues (Figure 1.2).

The diffusion rate along a specific direction varies depending on the observer's direction of looking. In other words, collecting dMRI data for any diffusion sensitization magnetic gradient allows the dMRI image intensity to be measured only along a single direction (Figure 1.3). Therefore, multiple dMR

4
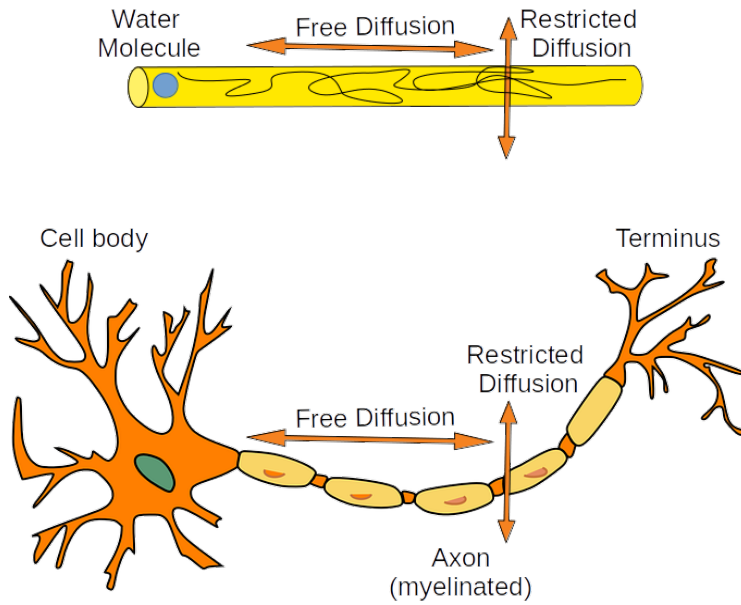
Figure 1.1: Water diffuses rapidly along the neuronal axons covered with myelin sheaths.
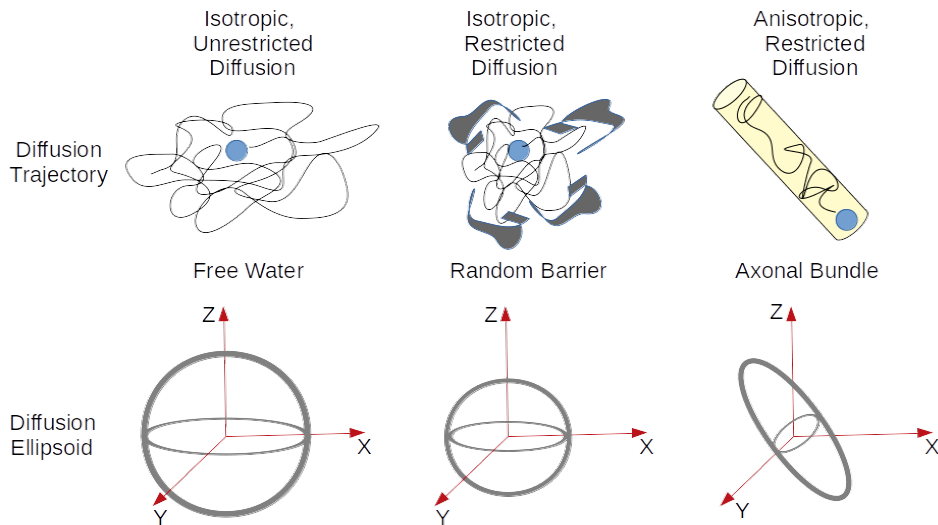
Figure 1.2: Different types of water diffusion interacting with barriers as well as their ellipsoid representation.
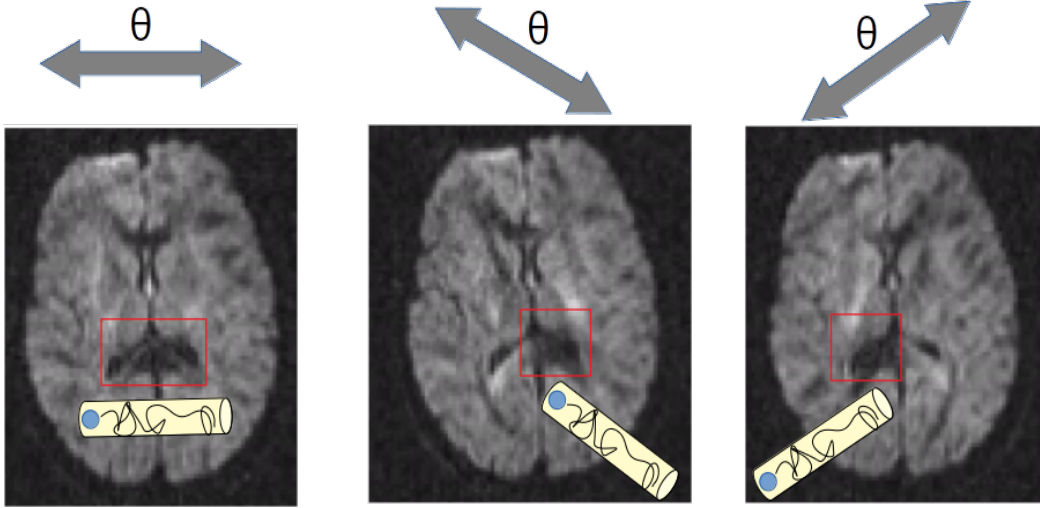
Figure 1.3: MR image intensity varies by applying different magnetic gradient directions.

images are collected sequentially by orienting the magnetic gradient along several directions for each location (i.e. voxel) in the brain. Each direction of magnetic gradient is denoted as a unit-norm vector $\theta \in \mathbb{R}^3$. The value of the measured signal depends on different parameters such as gradient strength, shown as scalar b, as well as the duration of radiation.

Given the gradient strength b and gradient direction θ for each voxel v, the dMRI signal can be estimated by the following equation [12, 35]:

$$\mathbf{S}(\theta, v) \approx \mathbf{w}_0 \mathbf{S}_0(v) e^{-A_0} + \sum_{f \in v} \mathbf{w}_f \mathbf{S}_0(v) e^{-b\theta^T \mathbf{Q}_{f,v}\theta}, \qquad (1.1)$$

where scalar $\mathbf{S}_0(v) \in \mathbb{R}$ is the signal measured without the presence of a diffusion sensitization magnetic gradient, $A_0 \in \mathbb{R}$ is the isotropic diffusion in all directions, and $\theta^T \mathbf{Q}_{f,v}\theta > 0$ is the diffusion at direction θ generated by fascicle f within the voxel v. $\mathbf{Q}_{f,v} \in \mathbb{R}^{3\times3}$ is a symmetric positive-definite matrix, called diffusion tensor, enables a compact representation of diffusion signal [10]. The diffusion tensor $\mathbf{Q}_{f,v}$ is usually represented as a 3D-ellipsoid and defined as:

$$\mathbf{Q}_{f,v} = \begin{bmatrix} \mathbf{u}_1 & \mathbf{u}_2 & \mathbf{u}_3 \end{bmatrix} \begin{bmatrix} s_a & 0 & 0 \\ 0 & sr_1 & 0 \\ 0 & 0 & sr_2 \end{bmatrix} \begin{bmatrix} \mathbf{u}_1^T \\ \mathbf{u}_2^T \\ \mathbf{u}_3^T \end{bmatrix} \qquad (1.2)$$
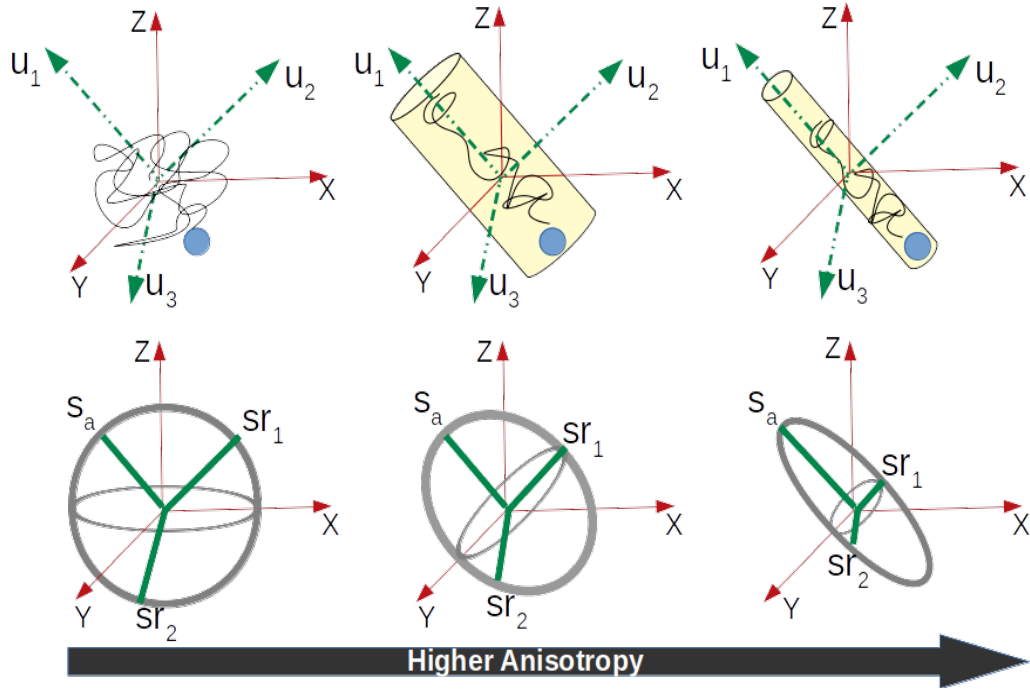
Figure 1.4: Representation of diffusion tensor as an ellipsoid as well as its eigenvalues including axial (s_a) and radial (sr_1 and sr_2) diffusivity.

where $\mathbf{u}_n \in \mathbb{R}^{3 \times 1}$ are unit-norm orthogonal vectors corresponding the local coordinate system in each voxel. s_a defines the axial diffusivity of the tensor showing the longitudinal direction of the ellipsoid aka the flow parallel to the axon. Both sr_1 and sr_2 correspond to the radial diffusivity of the tensor, which is the flow perpendicular to the axon and reflects the myelination around the it (Figure 1.4).

1.3   General Tractography Approaches

We know that the dMRI signal of the brain is the result of water diffusion through the bundles of axons, called fascicles. In general, all tractography algorithms try to solve the inverse problem of estimating the streamlines of fascicles given the diffusion information.

A simple tractography approach is called Streamline [61], which starts from one location, generates the next movement direction using the corresponding diffusion tensor to the current location (i.e. tracking process), and finally moves along the selected direction for a specific distance (i.e. step size). The

algorithm executes the tracking process and stepping repeatedly until a certain criterion is satisfied. If the generated tract meets certain anatomical criterion defined by an expert, it is accepted.

Typically, the result of a fiber tractography algorithm depends on multiple key factors [88]. The first factor corresponds to the validation criterion of the fiber tract, such as Fiber Orientation Distribution (FOD). This factor reflects the expert knowledge of the anatomical features related to the circuit under study. The second factor is the biological plausibility of generated fascicles. Overall, the fascicles must have smooth shapes with no abrupt directional changes. The third factor describes the movement direction selected in each step. This factor differentiates between probabilistic and deterministic tractography methods. The direction selection in a deterministic approach simply picks the peak direction of a diffusion tensor. However, the probabilistic tractography draws a sample stochastically from the distribution of all possible directions in the diffusion tensor, where each direction's chance of being selected is proportional to its magnitude in the directions' probabilistic distribution.

Therefore, each combination of these factors may result in different shapes of tracts.

1.4   Problem Statement

Standard practice in mapping connectomes from dMRI data is comprised of several steps: first acquire a dMRI image (Figure 1.5A), then fit a diffusion model to the signal in each brain voxel (Figure 1.5B) and finally use a tractography algorithm to estimate long range brain connections (Figure 1.5C). Multiple models can be used at each one of these steps and each model allows multiple parameters to be set. Currently, best practice in the field is to choose one diffusion model and pick a single set of parameters using heuristics such as recommendations by experts or previous publications. This human-based approach has several limitations. For example, different combinations of models and parameters generate different solutions and no one knows which one
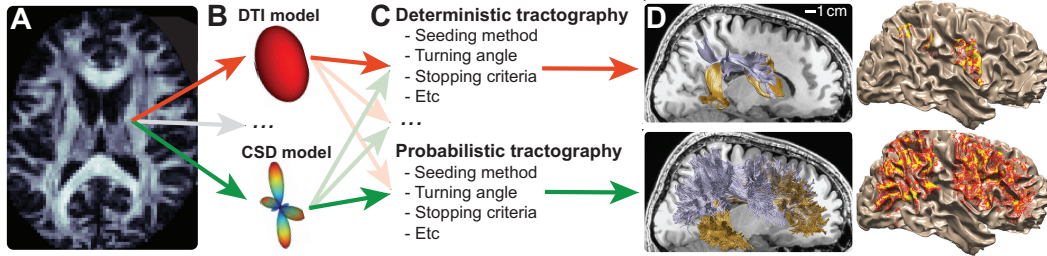
Figure 1.5: **A**: Measurements of white matter using diffusion-weighted magnetic resonance imaging (dMRI). **B**: Multiple models each can describe the dMRI signal in each brain voxel. For example, the diffusion-tensor model (DTI; top, [9]) and the constrained-spherical deconvolution model (CSD, bottom; [97]) are commonly used. **C**: Multiple tractography methods integrate model fits across voxels to estimate long-range brain connections. There are many tractography algorithms, each with multiple parameters, for both deterministic and probabilistic methods [96]. In principle several combinations of methods and parameters are used by investigators. **D**: **Left**: Two major white matter tracts, the arcuate fasciculus in gold and superior lateral fasciculus in lilac, reconstructed in a single brain using deterministic (top) and probabilistic (bottom) tractography. **Right**: Cortical termination of the superior lateral fasciculus in the same brain estimated with deterministic (top) and probabilistic (bottom) tractography. Throughout (A to D), arrows show multiple possible choices of model and parameters to generate connectome estimates (D) from dMRI data (A).

is the correct solution (Figure 1.5D). Figure 1.5 exemplifies how from a single dMRI data set collected in a brain, choosing a single model and parameters set (Figure 1.5A-C) can generate vastly different connectome mapping results (Figure 1.5D; adapted from [76]). In the figure, we show that both estimates of white matter tracts (Figure 1.5D left) and cortical connections (Figure 1.5D right) vary substantially even within a single brain.

There have been some supervised learning approaches proposed for tractography. These supervised methods, however, such as those using random forests [67] and neural networks [14, 77], require labelled data. This means tractography solutions must first be given for training, limiting the models mainly to mimic expert solutions rather than learn structures beyond them. A few methods have used regularized learning strategies, but for different purposes, such as removing false connections in the given tractography solution [28] and using radial regularization for micro-structure [24].

1.5 The Scope of this Project

Diffusion Tensor Imaging (DTI) and Constrained Spherical Deconvolution (CSD) are two types of DWI that apply various types of diffusion tensors to mathematically model the diffusion signals. For example, the DTI approach models the distribution of water molecules' spins using diffusion information as a Gaussian distribution, represented as a symmetric tensor matrix $\mathbf{Q}_{f,v} \in \mathbb{R}^{3\times3}$ with three diagonal elements corresponding the diffusion coefficient along each of the principal axis (x, y, z) as well as six off-diagonal parameters reflect the correlation of random motions between each pair of principal directions. More complex diffusion models represent the probability of each fiber tract along different directions, which enables representing of arbitrary fiber configurations such as crossing fibers (i.e. more than two fascicles crossing each other in a voxel). Then a tractography algorithm can be applied on diffusion tensors in order to infer the white matter connectivity of the brain, which means to determine which parts of the brain are connected to which other parts. The concentration of this project is to come up with an accurate data-driven tractography solution to infer the connection trajectories (i.e. fascicles) in the human brain given its diffusion data. Therefore, the remainder of this section discusses the mathematical representation of the input diffusion data. Also, we explain the output brain connectivity structure as a simple example for a voxel.

1.5.1 Input Data

For each voxel v, each element in the diffusion information vector $\mathbf{y}_v \in \mathbb{R}^{N_\theta}$ for N_θ magnetic gradient directions can be computed by Equation 1.1. Stacking all the vectors of $\mathbf{y}_v$ (for all voxels $v = 1, 2, 3, ..., N_v$) as columns gives us the diffusion matrix $\mathbf{Y} \in \mathbb{R}^{N_\theta \times N_v}$. Given the diffusion information $\mathbf{Y}$, we want to infer the structure of the brain connectomes as output. Later, in Section 2.1.1, we provide more details on the input data in our project.

1.5.2 Output Structure

In this section, we first discuss a representation of each fascicle considering its diffusion information, which is a basis for later mathematical encoding of the whole brain connectome described in Chapter 2. Then, we explain the inferred output structure of the brain using this representation.

Each fascicle in a voxel can be considered as a restricted barrier to the water diffusion in the presence of sensitization magnetic gradient. However, the diffusion direction changes along the curved structure of a fascicle. So different curvatures among a fascicle can be modeled by different diffusion tensors $\mathbf{Q}_a$ or ellipsoids with orientation a. Figure 1.6 demonstrates two crossing fascicles, f_1 and f_2, within a voxel v. In the segments of fascicles where the stream is straight enough in a voxel, the diffusion signal (aka ellipsoid) can be approximated by one orientation in the spatial space (e.g. f_2), but for the curved fascicles (e.g. f_1) the diffusion signal can be approximated by a linear combination of orientations. We discretize the whole spatial space into N_a predefined orientations and approximate each straight fascicle segment by an orientation vector. Our goal is to come up with the accurate chain of these orientations to approximate a fascicle structure the best, i.e. the correct set of orientations a for each fascicle-voxel pair. So for the pair of fascicle-voxel (f_1, v), the best orientation set is $\{a_2, a_3\}$, and for (f_2, v) is $\{a_1\}$.

Consequently, later in Chapter 2, we describe how to mathematically represent the whole structure of the brain connectome and how to formalize this problem in order to facilitate applying machine learning techniques.

1.6 Contributions

The ultimate goal of this research is to present a fully unsupervised learning framework for tractography. We exploit a recently introduced encoding for connectome data, called ENCODE [22], which represents dMRI (and white matter fascicles) as a tensor factorization. This factorization was previously used only to represent expert connectomes as a tensor, generated using a standard human-based tractography process introduced in Figure 1.5. We propose
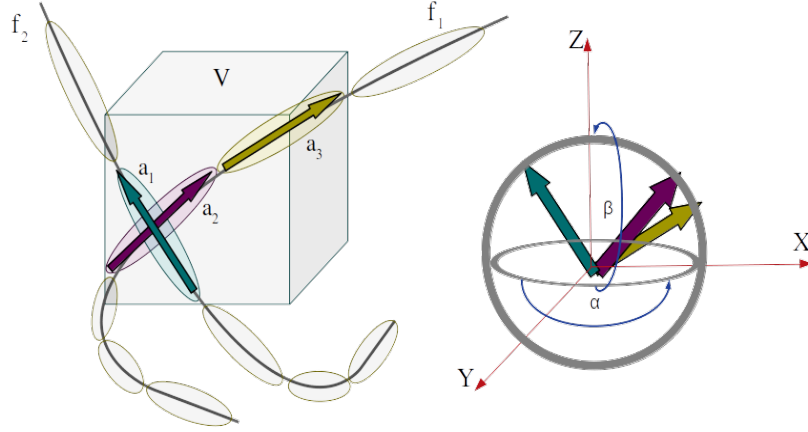
Figure 1.6: **Left**: Two crossing fascicles passing through a voxel are represented with multiple diffusion ellipsoids. Each ellipsoid can be approximated by an orientation in the space. **Right**: Discretizing the spatial space into the predefined orientation vectors.

to instead facilitate learning of this tensor using the dMRI data, to extract the structure of brain connectomes. In order to fully exploit machine learning algorithms to map brain connectome, it is convenient to encode connectome data (both dMRI and white matter fascicles) into multidimensional tensor frameworks that allow fast and efficient mathematical operations. We propose that tensor enconding allow extracting the structure of brain connectomes directly from the data.

We introduce the tensor encoding framework and show how it can be used in combination with regularization and optimization methods to overcome the limitations of the human-based standard tractography process introduced in Figure 1.5 to infer the connectome.

More specifically, we show that we can infer the structure of major white matter tracts (e.g. the Arcuate Fasciculus) directly from the data using group regularizers and convex optimization. We address two key challenges: (1) designing regularizers that adequately capture biologically plausible tract structures and (2) optimizing the resulting objective for an extremely high-dimensional and sparse tensor.

Tackling these challenges and introducing a data-driven method for map-

ping human brain connectome underlie the work in this thesis. Our contributions are:

1. we provide a way to (approximately) maximize the likelihood (i.e. the reconstruction loss) and provide an unconstrained objective function based on ENCODE to extract the tensor of the brain structure. This objective is highly under-constrained and ill-posed, and prefers any possible structures that accurately recreates the diffusion information.

   To come up with a constrained objective function, we introduce a group regularizer that attempts to extract a tensor that reflects a biologically plausible fascicle structure while also reconstructing the diffusion information. This group regularizer can capture both spatial and directional continuity of the white matter fascicles while enforcing the sparsity of the ultimate solution.

   We prove both that the objective is convex, and has a unique solution. Convexity ensures that gradient descent can obtain optimal solutions while the uniqueness guarantees achieving an identifiable solution.

2. To solve this extremely high-dimensional sparse problem, we use a greedy algorithm to screen the set of possible solutions upfront. We formally describe which dimension is the best option for shrinkage during the screening stage as well as how intuitively the selection criterion selects the best set of the possible solutions.

   We derive an efficient forward selection algorithm to solve each greedy step based on its previous step, in order to make the whole screening procedure efficient.

   We then provide approximation guarantees on the greedy algorithm to show that it has at least as good a submodularity ratio as a typical forward selection function.

   We show empirically that this greedy algorithm selects possible solutions much more effectively than a standard greedy algorithm, called Orthogonal Matching Pursuit (OMP).

3. We evaluate our proposed method on the synthetic data of five different datasets generated from publicly available diffusion weighted imaging (DWI) datasets. The idea is to study the generalization of our approach by designing various experiments on the dMRI data, (a) acquired via scanning devices with different configurations, (b) belonging to different human brain subjects, and (c) related to multiple tract structures. Our experiments enable us to see the impact of manipulating these three variables on the output solutions.

   We introduce different evaluation metrics and algorithms to measure the performance of each step in the whole procedure of extracting brain connectome. We show that how effectively the screening stage selects possible solutions, how well the optimization algorithm and group regularizer can capture the correct structure of fascicles, and how trustworthy the visualization algorithm is able to demonstrate the brain structure from multi-dimensional tensors.

   We show, both quantitatively and qualitatively, that the solutions provided by our method effectively reconstruct the diffusion information in each voxel while maintaining contiguous, smooth fascicles.

4. We provide an algorithm to visualize the solutions. Due to the information loss during the encoding of the connectome data, especially as the lost data corresponds to the positional information of the fascicles within voxels, it is not possible to retrieve the exact information. We introduce a visualization algorithm that optimizes among all the possible demonstration solutions in order to provide the most correct shape of tracts and their underlying fascicles based on biological properties of the neuronal bundles, in general.

1.7   Thesis Organization

The rest of this dissertation is organized to four chapters. Chapter 2 describes the required background on ENCODE as well as a short overview of previous

work done in the field to extract macroscopic brain structure. In Chapter 3, we discuss theoretical analysis to formulate our proposed tractography objective function. Then in Chapter 4, all methods and corresponding algorithms to optimize the proposed objective function are described. Chapter 5 provides empirical analysis of the designed experiments on different datasets. The proposed evaluation metrics and related algorithms are explained in details for both screening and optimization stages. We also discuss the visualization challenges and our proposed solution to tackle them. Finally, Chapter 6 concludes the thesis and describe some of the relative future research directions in the field [2].

---

[2]The code is available at: https://github.com/framinmansour/Learning-Macroscopic-Brain-Connectomes-via-Group-Sparse-Factorization

Chapter 2

Background and Related Work

This chapter puts together the required mathematical modeling background as well as related research done in the field. Section 2.1 describes the encoding framework in addition to the other required background for introducing our proposed method. Then Section 2.2 goes over some related works that apply machine learning techniques for the purpose of tractography. Section 2.3 discusses previous works that apply machine learning techniques to connectome data to analyse the subnetworks of the human brain. Connectome data could be either the functional MRI (fMRI) or diffusion MRI (dMRI), which are two neuroimaging modalities that allow in-vivo analysis of the brain network. Additionally, we also talk about some research projects trying to explore the applications of machine learning techniques in the clinical analysis of the human brain connectome.

2.1 Encoding Brain Connectomes as Tensors

ENCODE [22] maps fascicles from their natural brain space into the three dimensions of a sparse tensor $\underline{\mathbf{\Phi}} \in \mathbb{R}^{N_a \times N_v \times N_f}$ (Figure 2.1A - right). The first dimension of $\underline{\mathbf{\Phi}}$ (1st mode, size N_a) encodes individual white matter fascicles orientation at each position along their path through the brain. Individual segments (nodes) in a fascicle are coded as non-zero entries in the sparse array (dark-blue cubes in Figure 2.1A - right). The second dimension of $\underline{\mathbf{\Phi}}$ (2nd mode, size N_v) encodes fascicles' spatial position within the voxels of dMRI data. Slices in this second dimension represent single voxels (cyan slice in

A. Natural brain space and tensor encoding
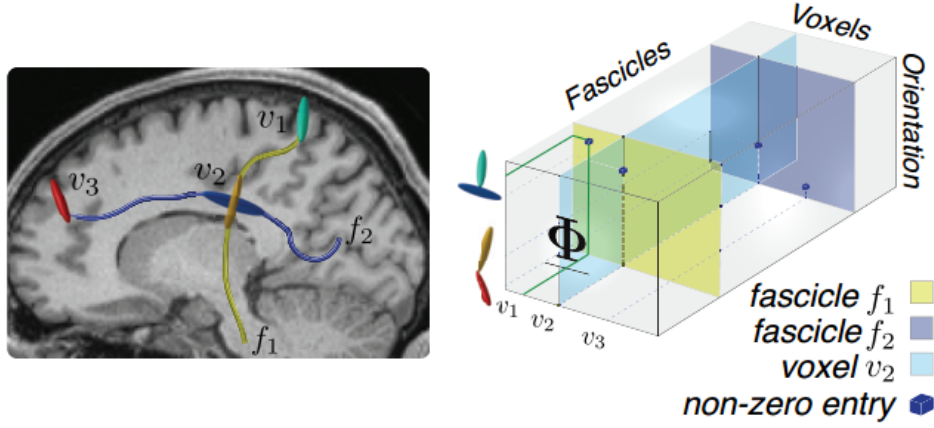
(a)

B. Model formulation and block reguralizers
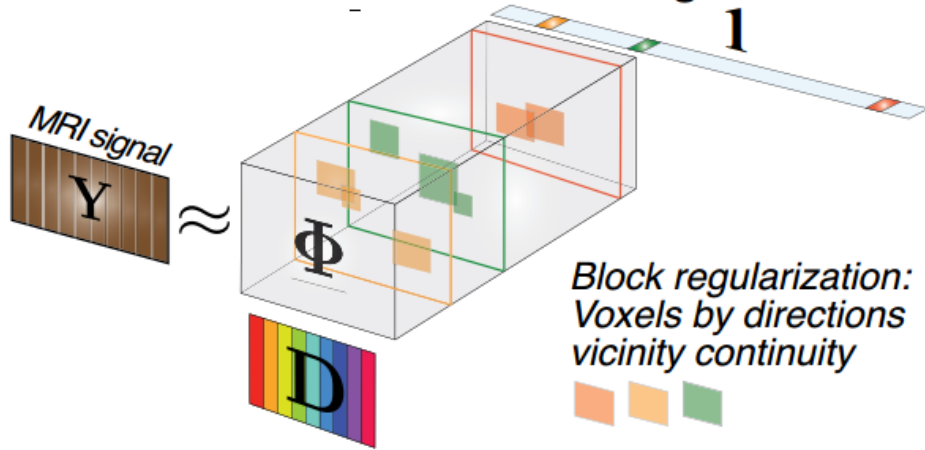
(b)

Figure 2.1: **A**: The ENCODE method; from natural brain space to tensor encoding. **Left**: Two white matter fascicles: f_1 passing through a sequence of voxels, including two voxels (v_1 and v_2), and f_2 passing through v_2 and v_3. **Right**: Encoding of the two fascicles in a three dimensional tensor. $\underline{\mathbf{\Phi}}$, whose non-zero entries indicate fascicles Orientation (1st mode), position (voxel-id, 2nd mode) and identity (3rd mode). **B**: Model formulation and group sparse regularization. Depiction of how ENCODE facilitates integration of dMRI signal $\mathbf{Y}$, connectome structure, $\underline{\mathbf{\Phi}}$, and a dictionary of predictions of the dMRI signal $\mathbf{D}$, for each fascicle orientation. The group regularizers (orange and green squares) define pairwise groups of neighbouring voxels and similar orientations. Note that the voxels are linearized to enable $\underline{\mathbf{\Phi}}$ and the groups to be visualized as this flattens the four-dimensional hyper-cubes—three dimensions for voxels and one for orientations—to squares.

17

Figure 2.1A - right). The third dimension (3rd mode, size N_f) encodes the indices of each fascicle within the connectome. Full fascicles are encoded as $\underline{\mathbf{\Phi}}$ frontal slices (cf., yellow and blue in Figure 2.1A - right). Within one tract, such as the Arcuate Fasciculus, the model we use has $N_a = 1057$ fine-grained orientations, with number of fascicles $N_f = 868$ and number of voxels $N_v = 11,823$.

ENCODE facilitates the integration of measured dMRI signals with the connectome structure (Figure 2.1B - right). dMRI measurements are collected with and without a diffusion sensitization magnetic gradient and along several gradient directions or N_θ, i.e. for some vectors $\theta \in \mathbb{R}^3$. In our data for Arcuate Fasciculuse for instance, the data includes $N_\theta = 96$ different angles of gradient Direction.

Thus, the dMRI signal is represented as matrix $\mathbf{Y} \in \mathbb{R}^{N_\theta \times N_v}$, which represents the value of diffusion signal received from each voxel when any individual angle of gradient directions was applied during the scanning.

Moreover, ENCODE allows factorizing the dMRI signal as the product of a 3-dimensional tensor $\underline{\mathbf{\Phi}} \in \mathbb{R}^{N_a \times N_v \times N_f}$ and a dictionary of dMRI signals $\mathbf{D} \in \mathbb{R}^{N_\theta \times N_a}$:$\mathbf{Y} \approx \underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1}$. The notation "$\times_n$" is the tensor-by-matrix product in mode-n (see [52]). The dot product with $\mathbf{1} \in \mathbb{R}^{N_f}$ sums over the fascicle dimension.[1]

The matrix $\mathbf{D}$ is a dictionary of representative diffusion signals: each column represents the diffusion signal we expect to receive from any axon in the direction of any possible fascicle orientation a by sensitizing magnetic gradient in each direction of θ, whose entries are $\mathbf{D}(\theta, a) = e^{-b\theta^T Q_a \theta} - \frac{1}{N_\theta} \sum_\theta e^{-b\theta^T Q_a \theta}$, where Q_a is an approximation of diffusion tensor per fascicle-voxel and scalar b denotes the diffusion sensitization gradient strength. Note $\theta^T Q_a \theta$ gives us the diffusion at direction θ generated by fascicle f. The values of b and Q_a are both given typically based on the expert knowledge.

---

[1]The original encoding uses a set of fascicles weights $\mathbf{w} \in \mathbb{R}^{N_f}$, to get $\mathbf{Y} \approx \underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{w}$. For a fixed $\underline{\mathbf{\Phi}}$, $\mathbf{w}$ was optimized to adjust the magnitude of each fascicle dimension. We do not require this additional vector, because these magnitudes can be incorporated into $\underline{\mathbf{\Phi}}$ and implicitly obtained when $\underline{\mathbf{\Phi}}$ is inferred.

2.1.1 Assumptions of our problem

Both matrices of the diffusion signal $\mathbf{Y}$ and dictionary $\mathbf{D}$ are given in this project. Also we assume that fascicles are fixed meaning that we know which fascicles are active for each voxel in tensor $\underline{\mathbf{\Phi}}$ that is, we know f_1 goes through voxels v_1 and v_2 and f_2 goes through v_2 and v_3 (Figure 2.1a). Our goal is to optimize $\underline{\mathbf{\Phi}}$ only on orientations to extract accurate fascicles' structure. This means that we want to infer which orientations are activated for f_1 in each of the voxels v_1 and v_2.

In a dictionary learning problem, the hidden variable is called *atom*. In our case, orientation play the role of the hidden variable. Moreover, in the representation of each fascicle segment as a diffusion ellipsoid, a *node* with spatial coordinate (x, y, z) can be associated to each ellipsoid. Then the full graph of a fiber could be constructed by connecting these nodes sequentially to each other. Hence, we might use the terms atom, orientation, and node interchangeably in the upcoming chapters when we discuss our proposed method.

2.2 Applying Machine Learning for Tractography

Poulin *et al.* proposed DeepTracker [77], that applies deep learning techniques, such as Recurrent Neural Networks (RNN) and Feed-Forward Neural Network (FFNN), to learn a model that can generate fiber streamlines directly from DWI data. FFNN serves as a local tractography model and has similar limitations and weaknesses to tractography models in learning the streamlines. On the other hand, RNN contributes whole data sequences and learns tracts globally. As a result, RNN model is more flexible for learning crossing fascicles. The training input data to each network is the normalized diffusion-weighted images distributed equally based on different gradient directions. The label is intuitively a streamline represented as a sequence S of $|S| = M$ coordinates $P_i(x_i, y_i, z_i)$ so that all consecutive pairs of points, P_i and P_{i+1} with specific direction d_i, are located at the same distance from each other. Therefore, the diffusion information at each location and its corresponding d_i direction label
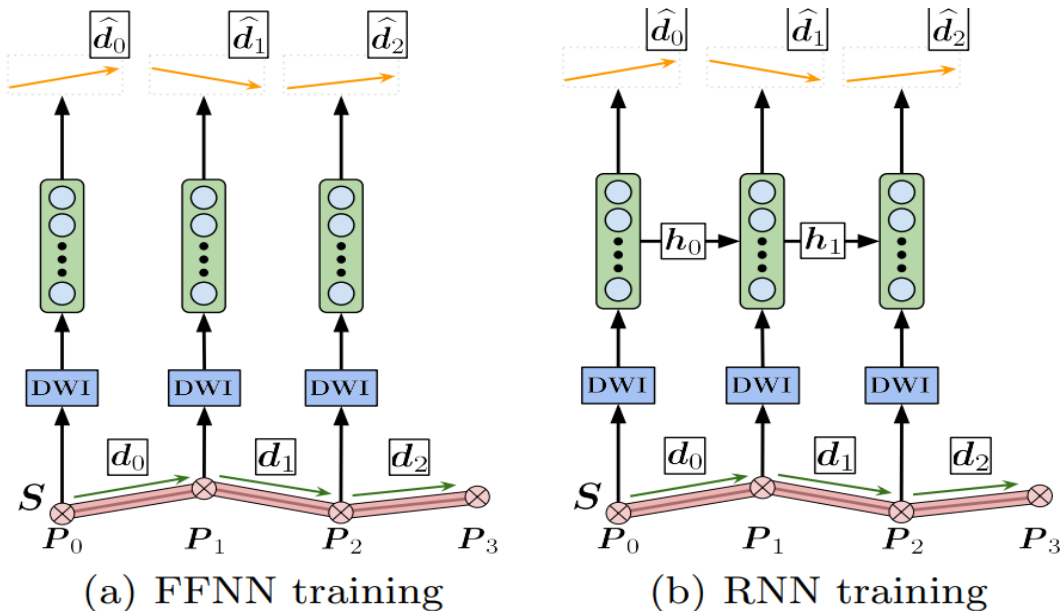
Figure 2.2: Architecture of FFNN and RNN models. (a) The diffusion information at data point P_i along with the stream S containing the ground-truth associated directions d_i (green) are inputted to the network. The output is the predicted directions $\hat{d}_i$ (orange). (b) RNN architecture allows sending information to itself in each step.

will be passed to the network during the training time. The output of the network would be a set of predicted directions $\hat{d}_i$, which would be compared against the ground-truth associated direction d_i (Figure 2.2 adapted from [77]).

An advantage of DeepTracker is that, unlike most of the tractography methods that require a diffusion model on the DWI data, it does not need any assumption on the diffusion data, which reduces user intervention. A disadvantage of DeepTracker is that, since it is a supervised learning method, it is highly dependent on the expert choices of directions. Therefore, the final structure of the connectome could only be as good as the expert generated structure. Moreover, it only allows deterministic tractography since the network output is a constant direction instead of a probabilistic distribution of multiple directions.

Benou and Raviv in [13] proposed a deep learning architecture that was identified to DeepTracker, called DeepTract. The only difference is that, instead of using RNN for a regression task to predict continuous fiber directions,

it instead learns a classifier that predicts probabilities of different predefined direction classes. They defined 724 direction classes by discretizing the 3D spatial space with angular resolution of $\sim 3.4°$. The advantage of using a classifier instead of a regressor is that it allows probabilistic tractography by providing the probability of each direction classes in the output. However, it is still a supervised learning method and limited to the given expert labels.

Jörgens *et al.* [49] focused on predicting a single step of streamline tractography using several predictors based on neural networks with the diffusion signal as their input. They designed several experiments with variety of configurations, and evaluated the performance of each setting on a dataset. They investigated different input scenarios and post-processing steps for both regression and classification tasks. Their results show that having the previous two directions as the input for predicting the next step of streamline tractography outperforms the prediction performance.

Similar to previous work, Wegmayr *et al.* in [105] also concentrated on predicting a single step tractography. They presented a neural network regression model that sequentially takes a local block of diffusion data as well as the last incoming direction of the fiber as input, and predict the next direction of the fiber as output. Both of these methods are supervised learning and only restricted to preform deterministic tractography.

In a few other works [58, 80, 104], the authors did not try to reconstruct fiber streamlines directly from DWI data. Instead, they proposed various deep learning methods, e.g. Convolutional Neural Networks (CNN) or U-Net CNN, to predict the appropriate diffusion model for the DWI data, which can later be used by a classical tractography algorithm to generate tract streamlines.

2.3 An Overview on Alternative Machine Learning Tools to Connectome Data

Previous work on connectome data represented the structural or functional connectivity of different brain networks, with nodes being regions and arcs being connections as binary or weighted edges in the network. Acquiring fMRI

and dMRI data of the brain is time-consuming and expensive. This means there are very few scans, especially when compared to the number of basic features in each connectome. This causes the problem of high dimensional small sample size (HDSSS) [65]. To address this problem for connectome data, it is typical to use feature selection, data augmentation, dimensionality reduction, and model regularization. Structural connectivity of the neurons in a human brain changes during the life time by learning, experiencing, aging, injury, and pathology [39]. We expect a unique structure for each individual brain, which probably requires a highly flexible and non-linear model to learn the connectivity of the brain over a population [60]. Furthermore, connectome data often has noisy labels and features since the labels rely on an expert's subjective measurement.

This section provides a review on the machine learning methods that take connectomes as input. Thus, it is essential to know about the general pipeline of constructing connectome from dMRI or fMRI data.

2.3.1 Connectome Data

A natural approach to represent the connectivity network of the brain formally is to use graphs. Each connectome can be represented as a graph $G(V, E)$, showing the connectivity between pairs of brain regions of interest (ROI) as a set of nodes V, connected with a set of either directed or undirected edges E. Then each connectome is typically encoded as an adjacency matrix, $A \in \mathbb{R}^{|V| \times |V|}$, with either weighted or binary entries showing the active connections between regions (i.e. which regions are connected to each other) [92]. The scale of the number of nodes, $|V|$, ranges from spatially sparse landmarks with a few nodes from large ROIs [6] to a densely distributed landmarks with many nodes from small ROIs [4]. Most of the previous works applied template atlases to define regions [99, 116]. However, other approaches try to apply unsupervised learning techniques to cluster regions based on features' similarities [69, 103], independent component analysis (ICA) [55], uniform tile-like parcellations of the brain, or hand-delineated ROIs. In an extreme case, a high scale $|V|$ could correspond to the number of voxels, which means that each individual voxel

considered to be a region. Unlike most of the previous works, we consider this extreme case for the connectome data since it hopefully results in a more precise brain structure.

Diffusion modeling is a way of characterizing the three-dimensional diffusion direction of water molecules as a function of spatial locations. It can describe the magnitude, the degree of anisotropy, and the orientation of diffusion anisotropy. Assigning appropriate weights to the edges between the pairs of ROIs is usually done by fitting a diffusion model such as DTI or CSD to each voxel, and then reconstructing the fibers via tractography algorithms [16]. Then the degree of connectivity could be defined as the number of tracts having the endpoints in both ROIs, or the probability that a tract connects two ROIs [36, 74], or the average of diffusion fractional anisotropy (FA) over all voxels containing a tract [60, 82]. We believe that using adjacency matrices, used by almost all of the previous works as the encoding approach of connectome data, has at least three major drawbacks for our learning purposes especially in extreme cases of $|V|$:

1. In general, this this representation does not provide any information about the structure of fascicles. There is no detailed information about the fascicles in each tract connecting two ROIs. Moreover, this representation is not able to encode any information about the crossing fascicles passing through each voxel. So the output structure of the brain connectome would not be precise in a voxel.

2. It stores redundant elements in the adjacency matrix, i.e. when there is a connection between two regions A and B, it stores the edges once from A to B and once from B to A. In the extreme cases when we consider all of the voxels, it would increase the usage of computational resources.

3. It is not possible to encode the exact information about the length of the tracts between two ROIs. Given the spatial coordinates of ROIs, we are only able to discriminate between a long tract and a short tract by approximating the distances between each pairs of ROIs. This would

make mathematical calculations on the matrices difficult, which prevents us from using the full potential of machine learning methods on this type of data.

Section 2.1 described another encoding brain connectome method to overcome the deficiencies of using adjacency matrices.

2.3.2 Machine Learning Tasks

Most of the machine learning tasks in this topic have focused on supervised classification and regression problems to predict a variety of neurological or psychological conditions as well as clinical health status of a subject. A few studies, however, concentrated on unsupervised learning methods to cluster patients' connectomes into different categories (e.g. to detect Alzheimer disease (AD) against normal control connectomes), or to select best ROI choices by optimization, or to identify important sub-networks [17]. We will discuss these tasks later in this section.

All these goals are totally different from our purpose in this project, which eventually is to be able to infer the connectome structure itself from dMRI data of the brain. However, some of the methods used in these works are relevant to our approach and so we will discuss them in the next sections.

A lot of machine learning research has dealt with class imbalance problems, choosing appropriate connectome features, learning models and kernels on labelled data, defining appropriate regularizers, and proving validation procedures, which are different from our task. Nonetheless, some of the machine learning models applied on connectome data such as methods of dimensionality reduction and feature selection as well as learning strategies are still partially relevant.

2.3.3 Feature Selection

A relevant scope to investigate in previous studies is about dimensionality reduction and feature selection methods, since we also work with huge tensors and matrices. From the literature, feature selection could be categorized in

three major groups:

Filters

This type of feature selector determines the importance of input features based on a heuristic criterion, and includes only the k most important features, for some k. This approach is usually efficient since it can be executed in parallel all over the dataset with respect to instances; however, it does not consider the impact of features' interactions when they combined together, i.e. is only able to consider the weights of input features independently [19].

Most of the works using this feature selection technique are based on applying t-test to find the significant differences over the distribution of edge features corresponding to two classes in order to find the most discriminative feature set [30, 48, 60, 73, 81, 117]. Additionally, the best choices of feature sets are the ones that are weakly correlated with others to avoid redundancy in the selected feature set [40].

Wrappers

Wrapper models identify the most important features for a prediction model, e.g. a classifier, as a preprocessing stage by fitting the model to a subset of training data. They select features and validate the selected subset simultaneously based on a measurement, e.g. for a classifier this measurement could be the accuracy or the error of the prediction model. One advantage of this approach is that it takes the combined discriminative power of selected feature sets with respect to the trained predictor into account. However, it is slower since it requires a model to be trained multiple times on different subsets of training data.

In general, wrappers consist of two groups of algorithms:

1. Backward sequential feature selection algorithms: Popular methods recursive feature elimination (REF) [38] and discriminative subnetwork mining (DSM) [33], start with the full set of features and remove them iteratively during the training of the model based on their elimination

impact on the value of error or accuracy in a classification task, respectively. Obviously, the fitting function that determines the elimination impact of a feature could be any arbitrary model rather than a classifier.

2. Forward sequential feature selection algorithms: In contrast to the backward selection methods, forward selection approaches start with an empty feature set and add to it iteratively based on the amount of improvement each feature produces in the value of fitting function. This value could be anything such as the error of a regression problem or the accuracy of a classifier. In Section 4.3, we will see that our proposed feature selection method could be categorized under forward selection wrappers.

Embedded selectors

In this approach, the procedure of feature selection is integrated in the training of final learned model by using regularizers in order to encourage a model that uses sparse subset of features.

2.3.4   Dimensionality Reduction

Here, we discuss dimensionality reduction methods used for connectome data in previous studies.

Instead of selecting the most essential features for a model in feature selection methods, dimensionality reduction techniques transform input feature space into a lower dimensional new representation space where the cardinal directions are designed to covary with the important components of the data [17].

Several papers applied popular dimensionality reduction methods such as principal component analysis (PCA) [18, 60, 78, 82, 118] or kernel based approaches [64].

One strategy to reduce the dimensionality of the feature space, $X \in \mathbb{R}^{N \times M}$, to the transformed space, $\hat{X} \in \mathbb{R}^{N \times \hat{M}}$, is through non-negative matrix factorization (NMF) [54], where $\hat{M}$ is the number of new features. In this method, $\hat{X}$ would be a weighting matrix of a basis representation, called dictionary,
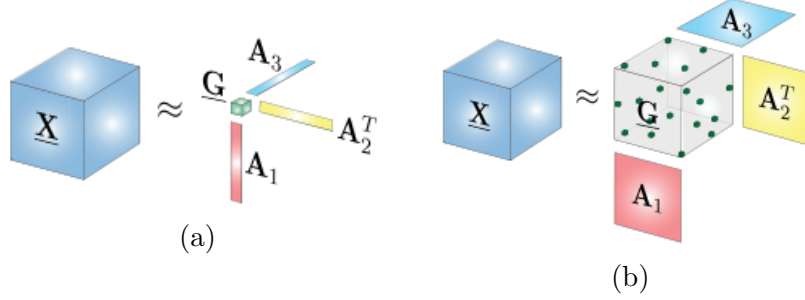
Figure 2.3: **(a):** Tucker decomposition **(b):** Sparse decomposition [21]

which is a high level non-negative feature transformation like $D \in \mathbb{R}^{\hat{M} \times M}$ that has to be learned along with $\hat{X}$. Then, decomposing original features into the new feature space is possible by $X = \hat{X} D + \epsilon$ through minimizing the residual error matrix ϵ. Non-negativity of the basis set D in this linear model enforces sparsity in the output weights $\hat{X}$. Therefore, selected features would be among the active entries in $\hat{X}$.

Tucker decomposition and sparse decomposition

This project builds up a model on top of a generalized version of a matrix decomposition method called Tucker decomposition [98]. In Tucker decomposition, a 3D tensor like $\underline{X} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ is approximated by the product of a core tensor $\underline{G} \in \mathbb{R}^{R_1 \times R_2 \times R_3}$ and factor matrices $A_n \in \mathbb{R}^{I_n \times R_n}$ for $n = 1, 2, 3, \ldots$:

$$\underline{X} \approx \underline{G} \times_1 A_1 \times_2 A_2 \times_3 A_3 \tag{2.1}$$

If $\underline{G}$ considered to be a very small dense tensor comparing to $\underline{X}$, meaning that $R_n \ll I_n$, then the new representation $\underline{G}$ guarantees data compression (Figure 2.3a adapted from [21]). Otherwise, if the core tensor $\underline{G}$ is a large sparse tensor, the decomposition is called sparse decomposition and data compression would still be achieved as long as $\underline{G}$ is sufficiently sparse (Figure 2.3b adapted from [21]).

Dimensionality reduction in general causes information loss. This lost information is often useful for visualizing discriminative features, particularly for connectome data where the anatomical locations of nodes or edges are informative for visualizing the learned model. Section 5.4 shows how we tackled this challenge in this project.

2.3.5 Learning Models

One of the barely-explored branches of machine learning applications on connectome data is *structured prediction*, where the output of the model is a structured object rather than categorical or continuous variables of a classifier or a regressor, respectively. This trend of research is the one most relative to our ultimate goal, which is to learn the structural connectivity of the human brain from the dMRI data without using any prior tractography knowledge.

Currently, many researchers are applying a wide variety of machine learning methods to connectome data using supervised methods including linear predictions, kernel based models, probabilistic models, ensembles, and stacked models such as deep neural networks [17]. Most of these works are irrelevant to our project either in terms of their purposes or their methodologies. They tried to train prediction models for classification or regression purposes from labeled data to diagnose disorders while our goal is to extract a brain connectome structure from unlabelled data. Therefore, we skip discussing these supervised learning approaches here.

A few works, however, applied unsupervised learning for different purposes such as clustering connectomes, identifying the discriminative subnetworks of a set of connectomes (e.g., identifying the discriminative subnetworks to differentiate between healthy control (HC) connectomes and post-traumatic stress disorders (PTSD) in a dataset [57, 109]), or to learn the strongly connected subnetwork structures in a single connectome. While some of these works tried to predict a specific structure in the connectome data, it is beneficial to overview unsupervised approaches.

Unsupervised learning methods

A subset of papers applied variations of dictionary learning methods for different purposes – many to predict subnetwork structures within or between connectomes. Li et al. applied a variation of general dictionary learning method, called Fisher discrimination dictionary learning (FDDL) [57, 109], in which the dictionary $D \in \mathbb{R}^{M \times K}$, with K Atoms (i.e. the basic elements to build the

dictionary) and M features, as well as the sparse encoded new representation of data $W \in \mathbb{R}^{K \times N}$, with N samples, would be learned through:

$$\{D^*, W^*\} = \underset{D,W}{\operatorname{argmin}} \mathcal{L}(X; D, W) + R(W) \qquad (2.2)$$

They applied the regularizer $R(W)$ to learn a *sparse W*. The loss function $\mathcal{L}$, a $L2$-norm, also encourages sub-dictionaries in D to include more discriminative and class specific information. Using this method, they learned two connectome subnetworks defined by columns of dictionary D, one for the PTSD and the other for HC,and then used these to distinguish PTSD versus HC connectomes. These sub-dictionaries can easily visualized since they contain edge information in order to demonstrate which parts of the brain are involved. Yoldemir et al. [111, 112] proposed a method called ''stable overlapping replicator dynamics'' (SORD) designed to produce the important subnetworks with strong connections within an individual connectome. By optimizing the following equation, they successfully determined the strong connections via node weights, w:

$$\mathrm{w}^* = \underset{\mathrm{w}}{\operatorname{argmax}} \mathrm{w}^T A \mathrm{w}$$
$$\text{s.t. } ||\mathrm{w}||_1 = 1 \text{ and } size(\mathrm{w}) \geq 1, \qquad (2.3)$$

where $A \in \mathbb{R}^{|V| \times |V|}$ is the adjacency matrix of the connectome. They decided about the strength of a connection by iteratively adding a new node and new edges to the A each time and optimizing the weights again. Thus, at each iteration, a subnetwork with strongest connections could be identified.

Given the high dimensionality of the connectome data, the complexity of the brain structure and its changes across time and among the population for different brain subjects, unsupervised learning methods, especially dictionary learning techniques, could provide appropriate ways to explore this area.

Chapter 3

Formalizing the Tractography Objective for Extracting Brain Connectomes

This chapter assembles the theoretical framework of our data-driven approach. Section 3.1 introduces an unconstrained optimization objective function by applying ENCODE and a simple maximum likelihood formulation. To form a constrained objective function with a unique solution to ensure an identifiable brain structure (Section 3.2), we design a group regularizer to capture the biological properties of a fascicle, i.e. continuity and smoothness of the fibers. We prove that the proposed objective function is convex and has a unique solution.

The original work on ENCODE assumed the tensor $\underline{\boldsymbol{\Phi}}$ was obtained from a tractography algorithm. In this chapter, we instead use this encoding to design an objective to infer $\underline{\boldsymbol{\Phi}}$ directly from dMRI data.

3.1 Unconstrained Objective Function

In this section, we discuss the general maximum likelihood formulation for multivariate distribution matrices, and then move forward by defining an unconstrained tractography objective function to reconstruct diffusion information.

3.1.1 Maximum Likelihood Formulation

We first consider a maximum likelihood formulation for the reconstruction loss, paralleling the loss functions considered for non-matrix data. This approach involves making distributional assumptions on the matrix $\mathbf{Y}$; we begin with the standard normal, though this could be generalized to other distributions—as is commonly done for generalized linear models—without eschewing convexity. Using the matrix normal distribution [71], we can formulate the loss between $\mathbf{Y}$ and the factorization by parameterizing the matrix normal using the factorized variables. Assume $\mathbf{Y} \sim \mathcal{MN}(\mathbf{M}, \mathbf{U}, \mathbf{V})$ where $\mathcal{MN}$ is the Multivariate Normal, $\mathbf{M}$ is the mean matrix, $\mathbf{U}$ is the row variance, and $\mathbf{V}$ is the column variance. As a common simplification, we will take $\mathbf{U} = \sigma_u^2 \mathbf{I}$ and $\mathbf{V} = \sigma_v^2 \mathbf{I}$. Then the pdf of $\mathbf{Y}$ is

$$
P(Y) = \frac{\exp\left(\frac{-1}{2}\mathrm{tr}(\mathbf{V}^{-1}(\mathbf{Y}-\mathbf{M})^\top \mathbf{U}^{-1}(\mathbf{Y}-\mathbf{M}))\right)}{(2\pi)^{N_\theta N_v/2}|\mathbf{V}|^{N_\theta/2}|\mathbf{U}|^{N_v/2}}
$$
$$
= \frac{\exp\left(\frac{-1}{2}\sigma_v \sigma_u \mathrm{tr}((\mathbf{Y}-\mathbf{M})^\top(\mathbf{Y}-\mathbf{M}))\right)}{(2\pi)^{N_\theta N_v/2}N_v N_\theta \sigma_v^{N_\theta} \sigma_u^{N_v}}
$$

because $|\mathbf{U}| = |\sigma_u^2 \mathbf{I}| = N_v \sigma_u^2$. This type of modeling approach assumes zero-mean, independent noise across entries in $\mathbf{Y}$, though with potentially different variance across voxels and across directions, which is a common assumption [12]. Taking the negative of the log likelihood, and dropping constants that do not affect the minimum, we obtain the optimization

$$
\operatorname*{argmin}_{\mathbf{M}} \mathrm{tr}((\mathbf{Y}-\mathbf{M})^\top(\mathbf{Y}-\mathbf{M}))
$$
$$
= \operatorname*{argmin}_{\mathbf{M}} \|\mathbf{Y}-\mathbf{M}\|_F^2,
$$

where $\|\cdot\|_F$ is the Frobenius norm that sums up the squared entries of the given matrix. tr or the trace of a square matrix is defined to be the sum of elements on the main diagonal of the matrix.

3.1.2 Unconstrained Tractography Objective Function

In order to define an unconstrained objective function, first consider the problem of estimating tensor $\underline{\boldsymbol{\Phi}}$ to best predict $\mathbf{Y}$, for a given $\mathbf{D} \in \mathbb{R}^{N_\theta \times N_a}$. A

standard maximum likelihood approach leads to the following reconstruction objective

$$\hat{\underline{\mathbf{\Phi}}} = \operatorname*{argmin}_{\underline{\mathbf{\Phi}} \in \mathbb{R}^{N_a \times N_v \times N_f}} \|\mathbf{Y} - \underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1}\|_F^2. \tag{3.1}$$

This objective prefers $\underline{\mathbf{\Phi}}$ that can accurately recreate the diffusion information in $\mathbf{Y}$. This optimization, however, is highly under-constrained, with many possible (dense) solutions.

In particular, this objective alone does not enforce a biologically plausible fascicle structure in $\underline{\mathbf{\Phi}}$. The tensor $\underline{\mathbf{\Phi}}$ should be highly sparse, because each voxel is expected to have only a small number of fascicles and orientations [76]. For example, for the Arcuate Fasciculus with $N_a \approx 120000$ and $N_f \approx 1000$, we expect at most an activation level in $\underline{\mathbf{\Phi}}$ of $N_v \times 10 \times 10/(N_a \times N_v \times N_f) \approx 1e{-}6$, using a conservative upper bound of 10 fascicles and 10 orientations on average per voxel. Additionally, the fascicles should be contiguous and should not sharply change orientation.

3.2 Constrained Objective Function

We design a group regularizer to encode these properties: sparsity, continuity, and smoothness of the fascicle structure in $\underline{\mathbf{\Phi}}$. Anatomical consistency of fascicles is enforced locally within groups of neighboring voxels and orientations. Overlapping groups are used to encourage this local consistency to result in global consistency. Group regularization prefers to zero all coefficients for a group. This zeroing has the effect of clustering non-zero coefficients, the entries of $\underline{\mathbf{\Phi}}$, in local regions within the tensor, ensuring similar fascicles and orientations are active based on spatial proximity. Further, overlapping groups encourages neighbouring groups to either both be active or both be inactive for a fascicle and orientation. This promotes contiguous fascicles and smooth direction changes. These groups are depicted in Figure 2.1B, with groups defined separately for each fascicle (slice). We describe the group regularizer more formally in the remainder of this section (Figure 3.1).
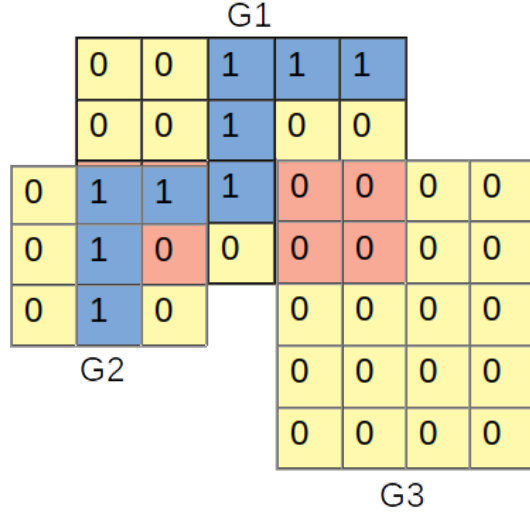
Figure 3.1: The intuition behind the group regularizer only for groups of voxels G_i. Each cell correspond to a voxel, where the blue active cells with value 1 show the voxels that share the same fascicle. G_2 overlaps the non-zero values in G_1 and prefers to become non-zero. G_3 overlaps the zero values in G_1 and prefers to stay zero in all entries.

3.2.1 Group regularizer

Assume we have groups of voxels $\mathcal{G}_\mathcal{V} \in \mathcal{V}$ based on spatial coordinates and groups of orientations $\mathcal{G}_\mathcal{A} \in \mathcal{A}$ based on orientation similarity. Note $\mathcal{V}$ is the set of groups of voxels and $\mathcal{A}$ is the set of groups of orientations. For example, each $\mathcal{G}_\mathcal{V}$ could be a set of 27 voxels in a local cube; these cubes of voxels can overlap between groups, such as $\{(1,1,1),(1,1,2),\ldots,(3,3,3)\} \in \mathcal{V}$ and $\{(2,1,1),(2,1,2),\ldots,(4,3,3)\} \in \mathcal{V}$. Each $\mathcal{G}_\mathcal{A}$ can be defined by selecting one atom (one orientation) and including all orientations in the group that have a small angle to that central atom, i.e., an angle that is below a chosen threshold (Figure 3.2).

The key intuition given by Figure 2.1B is that local blocks are defined for each fascicle (slice), in terms of all pairwise combinations of spatially close voxels $\mathcal{G}_\mathcal{V}$ and similar orientations $\mathcal{G}_\mathcal{A}$. Consider one (orientation, voxel, fascicle) triple (a, v, f). Assume a voxel has a non-zero coefficient for a fascicle, i.e. $\underline{\mathbf{\Phi}}_{a,v,f}$ is not zero for some a. A nearby voxel (here, a voxel within the same group $\mathcal{G}_\mathcal{V}$) is likely to have the same fascicle with a similar orientation. A

33
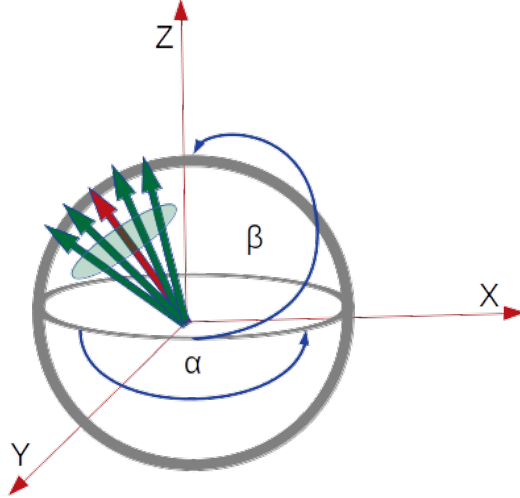
Figure 3.2: The way the groups of orientations are chosen.

distant voxel, on the other hand, is highly unlikely to share the same fascicle. The goal is to encourage as many pairwise groups $(\mathcal{G}_\mathcal{V}, \mathcal{G}_\mathcal{A})$ to be inactive—have all zero coefficients for a fascicle—and concentrate activation in $\underline{\mathbf{\Phi}}$ within groups. Note that the group regularizer is with respect to all $\mathcal{G}_\mathcal{V} \in \mathcal{V}$ and all $\mathcal{G}_\mathcal{A} \in \mathcal{A}$.

By trying to force as many blocks to zero as possible, the optimization will prefer to select a non-zero coefficient in the same block containing (a, v, f). Therefore, to incur least penalty, the neighboring voxel should pick such a fascicle and orientation, because it will simply be adding another non-zero entry to a block that already has non-zero entries. If, on the other hand, it tried to select a different fascicle or orientation outside of $\mathcal{G}_\mathcal{A}$ that caused a different block to go from being all zero, to suddenly having a non-zero coefficient, it would incur a sharper penalty. The group regularizer, therefore, tries to ensure that non-zero coefficients are clustered in blocks, which we design to reflect the desired spatial properties. Of course, this choice could cause a significant decrease in reconstruction error of $\mathbf{Y}$, if so, this would override the penalty incurred from the regularization, which means the optimization can still choose to violate the group regularizer, because the group regularization encodes preferences, rather than hard constraints.

One natural question is how to select these groups. A beneficial property

34

of encoding this structure as preferences in the form of regularizers rather than hard constraints, is that the objective is more robust to misspecification. For this reason, we select a simple approach and cover the tensor space with a large number of local groups, without carefully designing these groups using expert knowledge. Even if the groups are not defined optimally—for example, there are too small or too large—the objective should still be able to find a balance between reconstruction error and the spatial preferences encoded in the block regularizer.

We can enforce this group sparsity by adding a regularizer to Equation (3.1). Let $x_{\mathcal{G}_\mathcal{A},v,f} \in \mathbb{R}$ be non-zero whenever the fascicle f is active for voxel v, for any orientation $a \in \mathcal{G}_\mathcal{A}$. Let $\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}$ be the vector composed of these identifiers for each $v \in \mathcal{G}_\mathcal{V}$, where we have a different vector $\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}$ defined for each pair of $\mathcal{G}_\mathcal{V}, \mathcal{G}_\mathcal{A}$. This vector corresponds to a block of neighboring voxels $\mathcal{G}_\mathcal{V}$, and we want one of two cases. Either we want the entire vector $\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}$ to be zero, meaning the fascicle is not active in any of the voxels $v \in \mathcal{G}_\mathcal{V}$ for the orientations $a \in \mathcal{G}_\mathcal{A}$. Or, we want more than one non-zero entry in this vector, meaning multiple nearby voxels share the same fascicle. This second criterion is largely enforced by encouraging as many blocks to be zero as possible, because each voxel will prefer to activate fascicles and orientations in already active pairs $(\mathcal{G}_\mathcal{V}, \mathcal{G}_\mathcal{A})$. As with many sparse approaches, we use an ℓ_1-regularizer to set entire blocks to zero. To encode a preference to set entire blocks to zero, therefore, we use

$$\sum_{\mathcal{G}_\mathcal{V} \in \mathcal{V}} \sum_{\mathcal{G}_\mathcal{A} \in \mathcal{A}} \|\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}\|_2, \tag{3.2}$$

where the outer two sums can be seen as an ℓ_1 norm across the vector of norm values containing $\|\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}\|_2$ for every $\mathcal{G}_\mathcal{V}, \mathcal{G}_\mathcal{A}$. $\|.\|_2$ is the ℓ_2-norm, which is the square root of the sum of the squared vector values.

A canonical example of a subset selection approach that uses ℓ_1 is Lasso, which uses an ℓ_1 norm on the parameter vector for feature selection. The ℓ_1 prefers entries in the parameter vector to be set to zero, resulting in feature selection.

For the block regularizer, in particular, as has been previously done for block sparsity [93], we can use an ℓ_1 across the blocks $\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}$

$$\sum_{f \in \mathcal{F}} \sum_{\mathcal{G}_\mathcal{V} \in \mathcal{V}} \sum_{\mathcal{G}_\mathcal{A} \in \mathcal{A}} \|\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}\|_2, \tag{3.3}$$

which is just (3.2), with a summation over $\mathcal{F}$. The outer sums can be seen as an ℓ_1 norm across the vector of norm values containing $\|\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}\|_2$. This encourages $\|\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}\|_2 = 0$, which is only possible if $\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f} = \mathbf{0}$.

Finally, we need to define a continuous indicator variable $\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}$ to simplify the optimization. A natural choice is to use an indicator function, that is 1 if a fascicle is active in a voxel for the given orientations, and 0 otherwise. This choice, however, would result in a discontinuous regularizer, which would complicate the optimization. Instead, we propose the following continuous, not just $\{0,1\}$, indicator

$$\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f} = [\|\underline{\mathbf{\Phi}}_{\mathcal{G}_\mathcal{A},v_1,f}\|_1, \ldots, \|\underline{\mathbf{\Phi}}_{\mathcal{G}_\mathcal{A},v_n,f}\|_1] \in \mathbb{R}^{|\mathcal{G}_\mathcal{V}| \times 1} \quad \text{for each } v_i \in \mathcal{G}_\mathcal{V} \tag{3.4}$$

3.2.2 The tractography objective

For the given fascicle f and voxel in $\mathcal{G}_\mathcal{V}$, we determine if that fascicle f is active, for an orientation in $\mathcal{G}_\mathcal{A}$, by summing the absolute value of the coefficients for that voxel and fascicle across the orientations in $\mathcal{G}_\mathcal{A}$, i.e., $\|\underline{\mathbf{\Phi}}_{\mathcal{G}_\mathcal{A},v,f}\|_1$. Note that though this inner sum $\|\underline{\mathbf{\Phi}}_{\mathcal{G}_\mathcal{A},v,f}\|_1$ uses an ℓ_1 norm, the goal is to identify if a fascicle is active for the voxel for a set of orientations, rather than to encourage sparsity. An entry in $\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}$ is 0 if fascicle f is not active for any values from $(\mathcal{G}_\mathcal{V},\mathcal{G}_\mathcal{A})$. Otherwise, the entry is proportional to the sum of the absolute coefficient values for that fascicle for orientations in $\mathcal{G}_\mathcal{A}$. Our proposed group regularizer is

$$R(\underline{\mathbf{\Phi}}) = \sum_{f \in \mathcal{F}} \sum_{\mathcal{G}_\mathcal{V} \in \mathcal{V}} \sum_{\mathcal{G}_\mathcal{A} \in \mathcal{A}} \|\mathbf{x}_{\mathcal{G}_\mathcal{A},\mathcal{G}_\mathcal{V},f}\|_2 = \sum_{f \in \mathcal{F}} \sum_{\mathcal{G}_\mathcal{V} \in \mathcal{V}} \sum_{\mathcal{G}_\mathcal{A} \in \mathcal{A}} \sqrt{\sum_{v \in \mathcal{G}_\mathcal{V}} \left(\sum_{a \in \mathcal{G}_\mathcal{A}} |\underline{\mathbf{\Phi}}_{a,v,f}| \right)^2}, \tag{3.5}$$

which, combined with equation (3.1), gives our proposed objective. Given the observed $\mathbf{Y}$, and the dictionary $\mathbf{D}$, find the $\underline{\boldsymbol{\Phi}}$ s.t.

$$\operatorname*{argmin}_{\underline{\boldsymbol{\Phi}} \in \mathbb{R}^{N_a \times N_v \times N_f}} \|\mathbf{Y} - \underline{\boldsymbol{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1}\|_F^2 + \lambda R(\underline{\boldsymbol{\Phi}}) \tag{3.6}$$

for regularization weight $\lambda > 0$. This objective balances between reconstructing diffusion data and constraints on the structure in $\underline{\boldsymbol{\Phi}}$. One intuitive interpretation of this formulation is that, while the first term entails proper fit to the measured diffusion data, the second constrains the structure of the final model to produce global fascicle shapes consistent with anatomical knowledge. The expectation is to reconstruct continuous and smooth fascicles. Given this objective, we can now consider how to obtain the optimal $\underline{\boldsymbol{\Phi}}$. Crucially, this objective is convex in $\underline{\boldsymbol{\Phi}}$ and has a unique solution, which we show in Theorem 1 in Section 3.3. Uniqueness ensures identifiable tractography solutions and convexity facilitates obtaining optimal solutions of the objective function. In Chapter 5, we examine whether this optimal solution is correct or not.

3.3   Convexity of the Tractography Objective

An important property of this objective is that it is convex and has a unique solution for $\underline{\boldsymbol{\Phi}}$ (up to permutation), as we show in Theorem 1. The convexity of the objective ensures that gradient descent can obtain optimal solutions, which is critical for both improving the objective and ensuring that accurate tractography solutions are extracted. The uniqueness is important, because it means the solution is identifiable. For tractography, we would like to identify the fascicle structure for an individual; for an objective with multiple equivalent solutions, it is not clear which solution to select, and reflects an impreciseness in the objective. Any solution for $\underline{\boldsymbol{\Phi}}$ will always be equivalent, up to permutations of the fascicles (frontal slices, see Figure 2.1A - left), but should not change which fascicles are shared by which voxels.

Theorem 1. *The objective in* (3.6) *is convex in* $\underline{\boldsymbol{\Phi}}$. *Further, if the defined blocks $\mathcal{A}$ and $\mathcal{V}$ cover all possible orientations and voxels in the sense that every*

v is included in at least one group $\mathcal{G_V}$ and every orientation a is included in at least one group $\mathcal{G_A}$, then (3.6) has a unique solution (up to permutation).

Proof. Because the sum of convex functions is convex, to show that (3.6) is convex, we simply need to show that each term in the objective is convex.

The first term $\|\mathbf{Y} - \underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1}\|_F^2$ is convex in $\underline{\mathbf{\Phi}}$ because $\|\mathbf{Y} - \mathbf{M}\|_F^2$ is convex in $\mathbf{M}$, and $\mathbf{M} = \underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1}$ is an affine transformation of $\underline{\mathbf{\Phi}}$. The composition of an affine function and a convex function is convex.

The second term is the sum of several functions of $\underline{\mathbf{\Phi}}$, which only consider subparts of $\underline{\mathbf{\Phi}}$. If each of these functions in the group regularizer is convex, then the regularizer is composed of the sum of convex functions and so is itself convex. Let $R_{\mathcal{G_A},\mathcal{G_V},f}(\underline{\mathbf{\Phi}}) = \|\mathbf{x}_{\mathcal{G_A},\mathcal{G_V},f}\|_2$. This function only changes when elements in $\underline{\mathbf{\Phi}}$ related to $\mathcal{G_A},\mathcal{G_V},f$ change, and is otherwise constant. However, since a constant function is convex, $R_{\mathcal{G_A},\mathcal{G_V},f}$ is convex in the entries of $\underline{\mathbf{\Phi}}$ that are ignored. Let $\underline{\mathbf{\Phi}}_{\mathcal{G_A},\mathcal{G_V},f}$ be the entries in $\underline{\mathbf{\Phi}}$ that give $\mathbf{x}_{\mathcal{G_A},\mathcal{G_V},f} = [\|\underline{\mathbf{\Phi}}_{\mathcal{G_A},v_1,f}\|_1, \ldots, \|\underline{\mathbf{\Phi}}_{\mathcal{G_A},v_n,f}\|_1]$ for $v_i \in \mathcal{G_V}$. We can consider $\|\mathbf{x}_{\mathcal{G_A},\mathcal{G_V},f}\|_2$ as a vector composition, of $g : \mathbb{R}^k \to \mathbb{R}^n$ and $h : \mathbb{R}^n \to \mathbb{R}$, where $g(\underline{\mathbf{\Phi}}_{\mathcal{G_A},\mathcal{G_V},f}) = \mathbf{x}_{\mathcal{G_A},\mathcal{G_V},f}$ and $h(\mathbf{x}) = \|\mathbf{x}\|_2$. The resulting composition is $h(g(\underline{\mathbf{\Phi}}_{\mathcal{G_A},\mathcal{G_V},f})) = \|\mathbf{x}_{\mathcal{G_A},\mathcal{G_V},f}\|_2$. Each g_i of the vector-valued function g is convex in $\underline{\mathbf{\Phi}}_{\mathcal{G_A},\mathcal{G_V},f}$ because it applies an ℓ_1 norm—which is convex—on a subset of $\underline{\mathbf{\Phi}}_{\mathcal{G_A},\mathcal{G_V},f}$. Further, h is convex in $\mathbf{x}$, and non-decreasing in each $\mathbf{x}_i = g_i(\underline{\mathbf{\Phi}}_{\mathcal{G_A},\mathcal{G_V},f})$, because h is a norm. Therefore, the composition $h(g(\cdot))$ is convex. Therefore, because $R_{\mathcal{G_A},\mathcal{G_V},f}$ is convex w.r.t. $\underline{\mathbf{\Phi}}_{\mathcal{G_A},\mathcal{G_V},f}$ and constant w.r.t. all other values in $\underline{\mathbf{\Phi}}$, we know that $R_{\mathcal{G_A},\mathcal{G_V},f}$ is convex in $\underline{\mathbf{\Phi}}$. Since $R(\underline{\mathbf{\Phi}}) = \sum_{f\in\mathcal{F}} \sum_{\mathcal{G_V}\in\mathcal{V}} \sum_{\mathcal{G_A}\in\mathcal{A}} R_{\mathcal{G_A},\mathcal{G_V},f}(\underline{\mathbf{\Phi}})$ is a sum of convex function, it is convex.

To show uniqueness, we need to show that the regularizer is strictly convex. Because the sum of a strictly convex and convex function is again strictly convex, the resulting objective is itself strictly convex and so must have a unique minimum. Each component of the regularizer only considers a subset of $\underline{\mathbf{\Phi}}$; however, as long as each possible entry in $\underline{\mathbf{\Phi}}$ is considered at least once in one of these blocks, then that component of $\underline{\mathbf{\Phi}}$ has a strictly convex regularizer

38

on it in the objective, because norms are strictly convex. $\qquad\square$

Unfortunately, this objective has an extremely high-dimensional $\underline{\mathbf{\Phi}}$, where for a non-negligible λ, will also be extremely sparse. To efficiently solve for $\underline{\mathbf{\Phi}}$, we propose a novel greedy algorithm, combined with subgradient descent, to solve for $\underline{\mathbf{\Phi}}$. We provide the details for this algorithm in the next chapter.

Chapter 4

Optimizing the Tractography Objective for Extracting Brain Connectomes

In this chapter, we discuss the requirements of having an efficient optimization of the objective. Section 4.1 discusses the attributes of an efficient algorithm to optimize the tractography objective. Then Section 4.2 motivates the need for designing a new screening algorithm for pruning orientations by establishing the shortcomings of a standard greedy algorithm, called Orthogonal Matching Pursuit (OMP). After that, Section 4.3 proposes a new forward selection algorithm, called GreedyOrientation, that selects a set of orientations for each voxel greedily before optimisation. We dig into more details about GreedyOrientation algorithm by providing an efficient mechanism to compute orientations for each voxel. Then we highlight the approximation guarantees for the proposed screening strategy.

We finish off this chapter by jumping into the optimization algorithm, in which we discuss sparse tensor factorization methods and their corresponding issues. Then we wrap up (Section 4.4) by explaining the details of computing subgradient of the objective as well as the pseudo-code of the optimization algorithm.
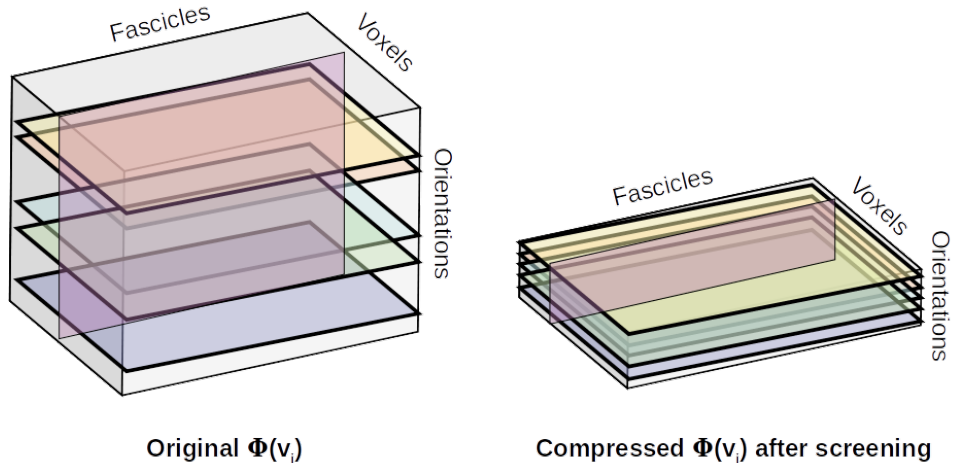
Figure 4.1: For each voxel, shrinking $\underline{\mathbf{\Phi}}$ by static screening

4.1 An Efficient Algorithm for the Tractography Objective

Standard gradient descent algorithms can be used directly on (3.6) to find the optimal solution. Unfortunately, the number of parameters in the optimization is very large: $N_v \times N_f \times N_a$ is billions even for just one tract. Fortunately, the number of *active* coefficients at the end of the optimization is much smaller, only on the order of N_v, because there are only handful of fascicles and orientations per voxel. Even when initializing $\underline{\mathbf{\Phi}}$ to zero, the gradient descent optimization might make all of $\underline{\mathbf{\Phi}}$ active during the optimization. Therefore, screening coefficients before running gradient descent is crucial.

Screening algorithms have been developed to prune entries for sparse problems [15, 107]. These generic methods, however, still have too many active coefficients to make this optimization tractable for wide application, as we have verified empirically.

Instead, we can design a screening algorithm specialized to our objective. The key is to define a suitable screening methodology, that does not introduce much approximation error into our objective. Without any screening, because our objective is convex, a simple gradient descent approach can obtain the global solution. Therefore, the main source of approximation error arise from the static screening (Figure 4.1).

Because each voxel can have a small number of fascicles, it similarly can only have a small number of orientations. Filtering fascicles a priori is difficult, because they contribute to a macrostructure and so need to be globally consistent. Plausible orientations for a voxel, however, can be defined locally and so are a natural dimension to screen before the optimization.

Orientations can largely be selected independently for each voxel, based solely on diffusion information. We can infer the likely orientations of fascicles in a voxel that could plausibly explain the diffusion information, without knowing precisely which fascicles are in that voxel. If we can select a plausible set of orientations for each voxel before optimizing the objective, we can significantly reduce the number of parameters. For example, 20 orientations is a large superset, but would reduce the number of parameters by a factor of 6,000 because the whole $N_a = 120,000$.

4.2   A Standard Greedy Algorithm for Screening

One strategy is to generate these orientations greedily, similar to the OMP (Orthogonal Matching Pursuit) method. This differs from most screening approaches, which usually iteratively prune starting from the full set. Generating orientations starting from an empty set, rather than pruning, is a more natural strategy for such an extremely sparse solution, which uses only 0.017% of the items, based on our results. Consider how OMP might generate orientations. For a given voxel v, the next best orientation is greedily selected based on how much it reduces the residual error for the diffusion. On the first step, it adds the single best orientation for predicting the $N_\theta = 96$ dimensional diffusion vector for voxel v. It generates up to a maximum of k orientations greedily and then stops. Then only coefficients for this set of orientations will be considered for voxel v in the optimization of the tractography objective. This procedure is executed for each voxel, and is very fast.

OMP   **GreedyOrientation**
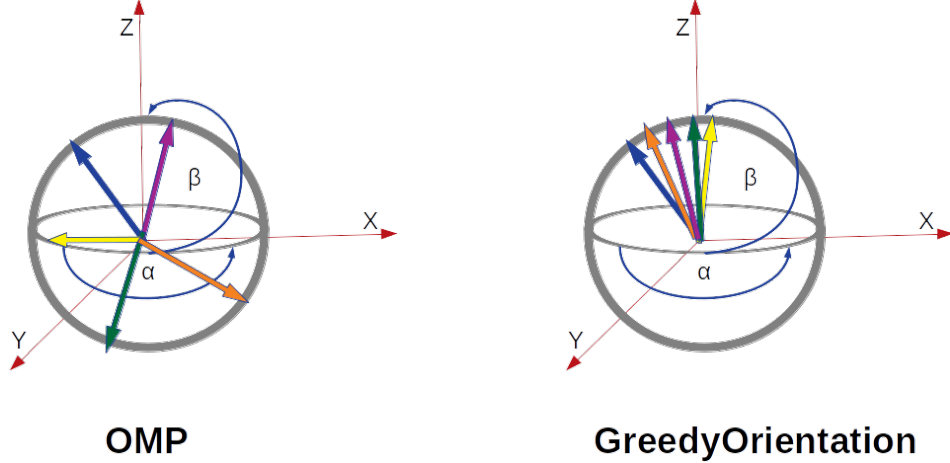
Figure 4.2: **Left**: OMP selects dissimilar orientations that maximize the gain. **Right**: GreedyOrientation selects similar orientations that maximize the gain.

4.2.1 Orthogonal Matching Pursuit

As shown in Algorithm 2, we define a criterion indicating the utility of an orientation (e.g., residual error for the voxel diffusion) and the greedy algorithm iteratively selects the orientation that most improves this criterion. The algorithm can generate a maximum of k such orientations, which for this project is $k = 5$ since there are few orientations per voxel.

Algorithm 1 Initialize $\underline{\mathbf{\Phi}}$ with optimized fascicle orientations
---
1: $\underline{\mathbf{\Phi}}\_\text{new} \leftarrow \underline{\mathbf{0}}_{N_a \times N_v \times N_f}$
2: // Get orientations for each voxel
3: **for** $v = 1, \ldots, N_v$ **do**
4: $\quad \underline{\mathbf{\Phi}}\_\text{new}(:, v, :) \leftarrow \text{OMP}(\mathbf{D}, \mathbf{\Phi}(:, v, :), \mathbf{y}(:, v))$
5: **Output:** $\underline{\mathbf{\Phi}}\_\text{new}$

Once the orientations are screened per voxel, the fascicle structure can be learned globally according to the block regularizer, but only enabling a reduced set of coefficients to become active.

4.2.2 Deficiencies of OMP for this problem

Though a greedy strategy for generating orientations is promising, the criterion used by OMP is not suitable for this problem. Using residual errors for the criterion prefers orthogonal or dissimilar orientations, to provide a basis with

43

Algorithm 2 select k orientations for each voxel

1: **procedure** OMP($\mathbf{D}, \mathbf{\Phi}, \mathbf{y}$)
2:      $s \leftarrow \emptyset$
3:      **while** not converge **do**
4:          $\mathbf{r} = \mathbf{y} - \mathbf{D}(:, s)\mathbf{\Phi}(s, :)\mathbf{1}$ $\triangleright$ Residue. $\mathbf{r} = \mathbf{y}$ if $s = \emptyset$
5:          $a \leftarrow \text{argmax}_{a \notin s} \frac{|\mathbf{r} \cdot \mathbf{D}(:, a)|}{||\mathbf{D}(:, a)||_2}$ $\triangleright$ Find the most correlated entry in the dictionary
6:          $\mathbf{\Phi}(a, :) \leftarrow rand(0, 0.0001)$ $\triangleright$ Randomly initialize the coefficients
7:          $s \leftarrow s \bigcup a$ $\triangleright$ Add to collection
8:      $\mathbf{\Phi}(s, :) \leftarrow \text{argmax}_{\mathbf{\Phi}(s, :)} ||y - \mathbf{D}(:, s)\mathbf{\Phi}(s, :)\mathbf{1}||_2^2$
9: **Output:** $\underline{\mathbf{\Phi}}(:, v, :)$

---

which to easily reconstruct the signal (Figure 4.2). The orientations in voxels, however, are unlikely to be orthogonal. Instead, it is more likely that there are multiple fascicles with similar orientations in a voxel, with some fascicles overlapping in different—but not necessarily orthogonal—directions. We must modify the selection criterion to select a number of similar orientations to reconstruct the diffusion information in a voxel.

4.3 Proposed Greedy Strategy for Screening

To modify the selection criterion, we rely on the more general algorithmic framework for subselecting items from a set, of which OMP is a special case. We will use a simple modification on Forward Selection for feature selection, which is very similar to OMP, and so will draw on the theoretical results for Forward Selection. Our goal is to define a set-function criterion $g(S)$, where S is a set of orientations that reflects the utility of an orientation for a voxel and for which simple, efficient greedy algorithms are effective. We need to define a criterion that evaluates the quality of subsets S from the full set of items $\underline{S}$. In our setting, $\underline{S}$ is the full set of orientations and S a subset of those orientations. We want to find $S \subset \underline{S}$ with $|S| \leq k$ such that $\bar{g}(S)$ is maximal.

If we can guarantee this criterion $\bar{g} : \mathcal{P}(\underline{S}) \rightarrow \mathbb{R}$ is (approximately) submodular[1], then we can rely on a wealth of literature showing the effectiveness

---

[1] A submodular function takes a set, informally, has the property that if we add a single element to its input, the difference in the incremental value of the function decreases as the

of greedy algorithms for picking S to maximize $\bar{g}$.

We use a simple modification on the criterion for OMP: the $g(S) =$ the squared multiple correlation [29]. This $g(S)$ corresponds to correlation coefficient for the given set of atoms S. The goal in Forward Selection is to maximize $g(S)$ for a given subset of orientations S such that $|S| \leq k$ for some maximum number of orientations k. Das and Kempe [29] show both that OMP and Forward Selection are using a greedy algorithm on an approximately submodular objective and, further, that the greedy algorithm for both Forward Selection and OMP is effective, both theoretically and empirically. We propose a simple yet effective modification, and define the GreedyOrientation criterion as

$$\bar{g}(S) \stackrel{\text{def}}{=} g(S) + \sum_{s \in S} g(\{s\})$$

This objective balances between preferring a set S with high multiple correlation, and ensuring that each orientation itself is useful. Each orientation likely explains a large proportion of the diffusion for a voxel. This objective will likely prefer to pick two orientations that are similar and that recreate the diffusion in the voxel well. This contrasts two orthogonal orientations, that can be linearly combined to produce those two orientation but that themselves do not well explain the diffusion information (Figure 4.2). This modification is conceptually simple, yet now has a very different meaning. The simplicity of the modification is also useful for the optimization, since a linear sum of submodular functions is itself again submodular and so we do not lose any of the submodularity properties.

The key fact property is a submodularity ratio, γ, that reflects the magnitude of the approximation error. Because we build on the Forward Selection criterion, we will have similar theoretical guarantees. We provide approximation guarantees for this submodular maximization in Section 4.3.2, using results for the multiple correlation [29].

---

size of the input set increases.

45

4.3.1 Derivation of Forward Selection for Orientations

To derive an efficient forward selection for the orientations, we need to make each greedy step efficient. For a fixed voxel v, the greedy algorithm selects the atom a that most increases $\bar{g}$:

$$\max_{a \notin S} \bar{g}(S \cup \{a\}) = \max_{a \notin S} g(S \cup \{a\}) + \sum_{s \in S} g(\{s\}) + g(\{a\}).$$

Note, we can compute $g(\{a\})$ upfront for each a and store it before doing the full greedy optimization. For the greedy optimization, the most naive solution would be to compute the full regression solution for each new subset $S \cup \{a\}$, to obtain $g(S \cup \{a\})$. Unfortunately, this brute-force approach is too expensive. Because of the structure of $\bar{g}$, however, we can take advantage of the solution on the previous step, to compute the solution on this step.

We provide the recursive update mechanism in Lemma 1. Let $\mathbf{y} \in \mathbb{R}^{N_\theta}$ be the diffusion information for one voxel. For given subset of orientations S with $|S| = k$, let $\mathbf{D}_S \in \mathbb{R}^{N_\theta \times k}$ be the subset of columns in $\mathbf{D}$ corresponding to orientations in S. Using similar subscript notation, with $a \notin S$ being a new atom not yet chosen in S, let

$$\mathbf{C}_S = \mathbf{D}_S^\top \mathbf{D}_S$$
$$\mathbf{b}_S = \mathbf{D}_S^\top \mathbf{y}$$
$$C_a = \mathbf{D}_a^\top \mathbf{D}_a$$
$$b_a = \mathbf{D}_a^\top \mathbf{y}$$
$$\mathbf{c}_{S,a} = \mathbf{D}_S^\top \mathbf{D}_a$$

where $D_a = D_{\{a\}}$. The squared multiple correlation is

$$g(S) = \mathbf{b}_S^\top \mathbf{C}_S^{-1} \mathbf{b}_S$$

and $g(\{a\}) = C_a^{-1} b_a^2$. We provide the following lemma to obtain an efficient mechanism to compute $g(S \cup \{a\})$ for each a. These recursive updates are similar to the updates given by [85], for OMP.

Lemma 1. *Given* $\mathbf{C}_S^{-1} \in \mathbb{R}^{k \times k}$, $\mathbf{b}_S$ *and* $g(S)$, *for*

$$\mathbf{c} = \mathbf{C}_S^{-1}\mathbf{c}_{S,a}$$

$$\nu = (C_a - \mathbf{c}_{S,a}^\top \mathbf{c})^{-1}$$

we get that

$$g(S \cup \{a\}) = g(S) + \nu(\mathbf{b}_S^\top \mathbf{c} - b_a)^2$$

Further

$$\bar{g}(S \cup \{a\}) = \bar{g}(S) + \nu(\mathbf{b}_S^\top \mathbf{c} - b_a)^2 + C_a^{-1}b_a^2$$

Proof. We know that $g(S \cup \{a\}) = \mathbf{b}_{S \cup \{a\}}^\top \mathbf{C}_{S \cup \{a\}}^{-1}\mathbf{b}_{S \cup \{a\}}$. We need to compute the inverse of $\mathbf{C}_{S \cup \{a\}}$ using the inverse of $\mathbf{C}_S$. We use the general block matrix inversion formula

$$\mathbf{C}_{S \cup \{a\}}^{-1} = \begin{bmatrix} \mathbf{C}_S & \mathbf{c}_{S,a} \\ \mathbf{c}_{S,a}^\top & C_a \end{bmatrix}^{-1}$$

$$= \begin{bmatrix} \mathbf{C}_S^{-1} + \nu\mathbf{C}_S^{-1}\mathbf{c}_{S,a}\mathbf{c}_{S,a}^\top\mathbf{C}_S^{-1} & -\nu\mathbf{C}_S^{-1}\mathbf{c}_{S,a} \\ -\nu\mathbf{c}_{S,a}^\top\mathbf{C}_S^{-1} & \nu \end{bmatrix}$$

Therefore,

$$g(S \cup \{a\}) = \mathbf{b}_{S \cup \{a\}}^\top \mathbf{C}_{S \cup \{a\}}^{-1}\mathbf{b}_{S \cup \{a\}}$$

$$= \mathbf{b}_S^\top\mathbf{C}_S^{-1}\mathbf{b}_S + \nu\mathbf{b}_S^\top\mathbf{C}_S^{-1}\mathbf{c}_{S,a}\mathbf{c}_{S,a}^\top\mathbf{C}_S^{-1}\mathbf{b}_S - 2\nu\mathbf{b}_S^\top\mathbf{C}_S^{-1}\mathbf{c}_{S,a}b_a + \nu b_a^2$$

$$= g(S) + \nu(\mathbf{b}_S^\top\mathbf{c})^2 - 2\nu b_a\mathbf{b}_S^\top\mathbf{c} + \nu b_a^2$$

$$= g(S) + \nu(\mathbf{b}_S^\top\mathbf{c} - \nu b_a)$$

Using this, we can see that

$$\bar{g}(S \cup \{a\}) = g(S \cup \{a\}) + \sum_{s \in S \cup \{a\}} g(\{s\})$$

$$= g(S) + \nu(\mathbf{b}_S^\top\mathbf{c} - \nu b_a)^2 + \sum_{s \in S} g(\{s\}) + g(\{a\})$$

$$= \bar{g}(S) + \nu(\mathbf{b}_S^\top\mathbf{c} - \nu b_a)^2 + g(\{a\})$$

completing the proof, because $g(\{a\}) = C_a^{-1}b_a^2$. $\qquad\square$

Given this result, the computation of $g(S \cup \{a\})$ for one atom a costs $O(kN_\theta + k^2) = O(kN_\theta)$, since $N_\theta > k$. To compute g for each a, therefore, costs a total of $O(kN_\theta N_a)$. We summarize the greedy algorithm for computing the directions for a voxel in Algorithm 3. A new point a is added to greedily maximize $\bar{g}(S)$, until S has k orientations.

Algorithm 3 GreedyOrientation: greedy algorithm to select directions for each voxel
___

1: **Input**: dictionary $\mathbf{D}$, maximum number of directions k, diffusion signal $\mathbf{y}$
2: **Output**: set of selected orientations S
3: // Compute $g(\{a\})$ for each a, $\mathbf{g} = \text{diag}(\mathbf{D}^\top\mathbf{D})^{-1}(\mathbf{D}^\top\mathbf{y})^2$
4: $\mathbf{c} \leftarrow \mathbf{D}^\top\mathbf{D}$
5: $\mathbf{b} \leftarrow \mathbf{D}^\top\mathbf{y}$
6: $\mathbf{g} \leftarrow \mathbf{0}$
7: $a_{\max} \leftarrow -1, g_{\max} \leftarrow 0$
8: **for** $a = 1, \ldots, N_a$ **do** $\triangleright\ O(N_\theta N_a)$
9:      $\mathbf{g}(a) \leftarrow (\mathbf{b}(a))^2/\mathbf{c}(a,a)$
10:     **if** $g_{\max} < \mathbf{g}(a)$ **then**
11:          $g_{\max} \leftarrow \mathbf{g}(a),\ a_{\max} \leftarrow a$

12: $S \leftarrow a_{\max}$
13: $\mathbf{C}^{-1} \leftarrow 1/\mathbf{c}(a_{\max}, a_{\max})$
14: **for** $i = 2, \ldots, k$ **do**
15:     // Compute $g(S \cup \{a\})$ for every a
16:     $\mathbf{g}_S, \nu \leftarrow \text{ComputeGain}(S, \mathbf{C}^{-1}, \mathbf{c}, \mathbf{b})$ $\triangleright\ O(kN_\theta)$
17:     $\bar{\mathbf{g}} \leftarrow \mathbf{g}_S + \mathbf{g}$ $\triangleright\ \bar{\mathbf{g}} \in \mathbb{R}^{N_a}$
18:     $a_{\max} \leftarrow \text{argmax}_{a\notin S}\ \bar{\mathbf{g}}(a)$ $\triangleright\ O(N_a)$
19:     $\mathbf{C}^{-1} \leftarrow \begin{bmatrix} \mathbf{C}^{-1} + \nu(a_{\max})\mathbf{C}^{-1}\mathbf{c}(S, a_{\max})\mathbf{c}(S, a_{\max})^\top\mathbf{C}^{-1} & -\nu(a_{\max})\mathbf{C}^{-1}\mathbf{c}(S, a_{\max}) \\ -\nu(a_{\max})\mathbf{c}_{S,a}^\top\mathbf{C}_S^{-1} & \nu(a_{\max}) \end{bmatrix}$
20:     $S \leftarrow S \cup \{a_{\max}\}$
21: **Output:** S

Algorithm 4 ComputeGain$(S, \mathbf{C}^{-1}, \mathbf{c}, \mathbf{b})$
___

1: **for** $a = 1, \ldots, N_a$ **do**
2:      $\tilde{\mathbf{c}} \leftarrow \mathbf{C}^{-1}\mathbf{c}(S, a)$
3:      $\nu(a) \leftarrow (\mathbf{c}(a, a) - \mathbf{c}(S, a)^\top\tilde{\mathbf{c}})^{-1}$
4:      $\mathbf{g}_S(a) \leftarrow \nu(a)(\mathbf{b}(S)^\top\tilde{\mathbf{c}} - \mathbf{b}(a))^2$
5: **Output:** $\mathbf{g}_S, \nu$

4.3.2 Theoretical Guarantees of the Greedy Screening Strategy

We can obtain approximation guarantees from the fact that the approximately submodular function for GreedyOrientation has at least as good a submodularity ratio as the typical forward selection function g. The submodularity ratio is defined as

$$\gamma(g) \stackrel{\text{def}}{=} \min_{(S,L)\in\mathcal{S},\ S\cap L=\emptyset} \frac{\sum_{y\in S}(g(L\cup\{y\})-g(L))}{g(L\cup S)-g(L)} \tag{4.1}$$

for non-negative functions $g : \mathcal{P}(\underline{\mathcal{S}}) \to \mathbb{R}^+$. According to the definition, a submodular function has the property that the difference in the incremental value of the function that a single element makes when added to an input set, decreases as the size of the input set increases. Therefore, if g is a monotone function and $\gamma(g) \geq 1$, then g is submodular. Otherwise, for $\gamma(g) < 1$, the function is not submodular and is instead called approximately submodular for $\gamma(g)$ close to 1, i.e. $1 - \epsilon < \gamma(g) < 1$. The closer $\gamma(g)$ is to 1, the better the approximation guarantees of greedy algorithms on these functions, with the best approximation guarantees for $\gamma(g) \geq 1$.

In the following theorem, we show that our GreedyOrientation function $\bar{g}$ has a submodularity ratio that is no worse than ForwardSelection. The proof highlights that in fact the ratio is likely strictly better.

Theorem 2. *If $g : \mathcal{P}(\underline{\mathcal{S}}) \to \mathbb{R}^+$ is a monotone function, and*

$$\bar{g}(S) \stackrel{\text{def}}{=} g(S) + \sum_{s\in S} g(\{s\})$$

then

$$\gamma(\bar{g}) \geq \gamma(g).$$

Proof. For clarity, we introduce notation for the numerator and denominator of the submodularity ratio:

$$\gamma_N(g, L, S) \stackrel{\text{def}}{=} \sum_{y\in S}(g(L\cup\{y\})-g(L))$$

$$\gamma_D(g, L, S) \stackrel{\text{def}}{=} g(L\cup S)-g(L)$$

49

Notice that

$$\bar{g}(L) = g(L) + \sum_{x \in L} g(\{x\})$$

$$\bar{g}(L \cup \{y\}) = g(L) + \sum_{x \in L} g(\{x\}) + g(\{y\})$$

$$\bar{g}(L \cup S) = g(L \cup S) + \sum_{x \in L} g(\{x\}) + \sum_{y \in S} g(\{y\}), \text{ where } S \cap L = \emptyset$$

giving

$$\gamma_N(\bar{g}, L, S) = \sum_{y \in S} (\bar{g}(L \cup \{y\}) - \bar{g}(L))$$

$$= \sum_{y \in S} (g(L \cup \{y\}) + g(\{y\}) - g(L))$$

$$= \gamma_N(g, L, S) + \sum_{y \in S} g(\{y\})$$

and

$$\gamma_D(\bar{g}, L, S) = \bar{g}(L \cup S) - \bar{g}(L)$$

$$= g(L \cup S) + \sum_{y \in S} g(\{y\}) - g(L)$$

$$= \gamma_D(g, L, S) + \sum_{y \in S} g(\{y\}).$$

If we let $a_S \stackrel{\text{def}}{=} \sum_{y \in S} g(\{y\}) \geq 0$, then we get that

$$\gamma(\bar{g}) = \min_{S, L \in \underline{\mathcal{S}}, S \cap L = \emptyset} \frac{\gamma_N(\bar{g}, L, S)}{\gamma_D(\bar{g}, L, S)}$$

$$= \min_{S, L \in \underline{\mathcal{S}}, S \cap L = \emptyset} \frac{\gamma_N(g, L, S) + a_S}{\gamma_D(g, L, S) + a_S}$$

$$\geq \min_{S, L \in \underline{\mathcal{S}}, S \cap L = \emptyset} \frac{\gamma_N(g, L, S)}{\gamma_D(g, L, S)} \qquad \triangleright \text{ because } a_S > 0 \text{ and } \gamma_D \geq \gamma_N$$

$$= \gamma(g)$$

completing the proof.

□

The following result now easily follows, from Theorem 4.2 [29], for approximately submodular functions.

Corollary 1. *The set of orientations S chosen by GreedyOrientation satisfies*

$$\bar{g}(S) \geq \left(1 - e^{-\gamma(\bar{g})}\right) OPT$$

where $OPT = \bar{g}(S^*)$ for the optimal selection S^* such that $|S^*| = k$.

4.4 Sparse Tensor Factorization Algorithm

In this section, we develop an algorithm to optimize our extremely sparse, high-dimensional objective. As mentioned earlier in this chapter, a common strategy for sparse optimization problems is to first perform screening on the coefficients — which corresponds to all of $\underline{\mathbf{\Phi}}$ in this setting — to avoid modifying coefficients that will remain zero. A number of generalized screening approaches have been developed for general sparse problems, either with a static screening before the start of the optimization [75] or with a dynamic screening that adjust the set of feasible coefficients during the optimization [15]. We proposed a specialized static screening, where we first select a set of feasible orientations for each voxel. This static screening on the entries in $\underline{\mathbf{\Phi}}$ significantly reduces the cost per iteration of gradient descent and reduces the number of iterations. This significantly speeds up the optimization, without incurring much approximation error, because of the approximation guarantees of the static screening approach. We first highlight why standard matrix and tensor factorization algorithms are not suitable for this problem, and then derive our specialized solver.

4.4.1 Issues with Using Standard Matrix or Tensor Factorization Algorithms

A natural approach to consider to obtain $\underline{\mathbf{\Phi}}$ is tensor factorization. Our goal is to factorize a matrix $\mathbf{Y}$ in a tensor $\underline{\mathbf{\Phi}}$, for a given dictionary $\mathbf{D}$, such that $\mathbf{Y} = \underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1}$. Much of the work in tensor factorization, however, has focused on decomposing tensors into a set of matrices, with a small core tensor with the Tucker decomposition focused on low rank tensor factorizations (see [25] for a thorough overview). A few of these works have examined how to

obtain a sparse core tensor, but towards the aim of either enforcing uniqueness [62] or to obtain core tensors that are more efficient to store and use [20, 26, 91, 119]. There has been some work on factorizing large sparse tensors, for tensor-SVD [2, 114]; again, however, their goal is to factorize a sparse tensor, which differs from our goal to factorize a dense matrix into an extremely sparse tensor with a particular structure. The most closely related algorithm is derived for low-rank regularizers for non-negative tensor factorization [**xu2013ablock**], but it is not designed for large sparse tensors.

4.4.2 Computing the Subgradient of the Objective

Once the orientations are set per voxel using GreedyOrientation, we can much more efficiently compute the gradient for the objective, because the sum over groups significantly reduces,

$$\sum_{\mathcal{G}_{\mathcal{V}} \in \mathcal{V}} \sum_{\mathcal{G}_{\mathcal{A}} \in \mathcal{A}} \|\mathbf{x}_{\mathcal{G}_{\mathcal{A}}, \mathcal{G}_{\mathcal{V}}, f}\|_2 = \sum_{\mathcal{G}_{\mathcal{V}} \in \mathcal{V}} \sum_{\mathcal{G}_{\mathcal{A}} \in \mathcal{A}(\mathcal{G}_{\mathcal{V}})} \|\mathbf{x}_{\mathcal{G}_{\mathcal{A}}, \mathcal{G}_{\mathcal{V}}, f}\|_2$$

where $\mathcal{A}(\mathcal{G}_{\mathcal{V}}) = \{\mathcal{G}_{\mathcal{A}} \in \mathcal{A} \mid \mathcal{G}_{\mathcal{A}} \cap S(v) \neq \emptyset, v \in \mathcal{G}_{\mathcal{V}}\}$. The set $\mathcal{A}(\mathcal{G}_{\mathcal{V}})$ only includes groups with orientations that are active for at least one voxel in $\mathcal{G}_{\mathcal{V}}$.

Let there be N_{G_a} atom groups and N_{G_v} voxel groups. We use $\mathbf{G}_A \in \{0, 1\}^{N_a \times N_{G_a}}$ (resp. $\mathbf{G}_V \in \{0, 1\}^{N_v \times N_{G_v}}$) to denote if an atom (resp. a voxel) belongs to a group. Specifically, if $\mathbf{G}_A(a, g) = 1$, then atom a belongs to group g; otherwise it does not. The subderivative of the group regularizer w.r.t. $\underline{\mathbf{\Phi}}(a, v, f)$ is

$$\lambda_g \sum_{a_g}^{N_{G_a}} \mathbf{G}_A(a, a_g) \sum_{v_g}^{N_{G_v}} \mathbf{G}_V(v, v_g) \frac{\sum\limits_{a_i \in \mathcal{G}_{a_g}} |\underline{\mathbf{\Phi}}(a_i, v, f)|}{\underline{\mathbf{A}}(a_g, v_g, f)} \operatorname{sign}(\underline{\mathbf{\Phi}}(a, v, f))$$

where tensor $\underline{\mathbf{A}} = \sqrt{\left(|\underline{\mathbf{\Phi}}| \times_1 \mathbf{G}_A^\top\right)^2 \times_2 \mathbf{G}_V^\top}$. Additionally, we include a standard ℓ_1 regularizer on all of $\underline{\mathbf{\Phi}}$ to further promote sparsity. The full subgradient of the objective w.r.t. $\underline{\mathbf{\Phi}}$ is

$$\begin{aligned}
\nabla_{\underline{\mathbf{\Phi}}} =& \mathbf{D}^\top (\underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1} - \mathbf{Y}) \mathbf{1}^\top \\
&+ \lambda_g \left(\left\{ \left(|\underline{\mathbf{\Phi}}| \times_1 \mathbf{G}_A^\top \right) \circ \left(\frac{1}{\underline{\mathbf{A}}} \times_2 \mathbf{G}_V \right) \right\} \times_1 \mathbf{G}_A \right) \circ \operatorname{sign}(\underline{\mathbf{\Phi}}) \qquad (4.2) \\
&+ \lambda_1 \operatorname{sign}(\underline{\mathbf{\Phi}}),
\end{aligned}$$

where $\circ$ shows the element wise matrix multiplication. On each step, we step in the direction of the negative of the gradient, with a fixed stepsize, which we set to $\eta = 1e - 3$, until the objective improvement is below a threshold $1e - 4$ or until a maximum number of iterations is reached. In addition to the initial screening, we can obtain some speed improvements on each step by only computing the gradient for currently active elements in $\underline{\Phi}$. For any zero-ed elements in $\underline{\Phi}$, the gradient of the regularizer would be zero, as the regularizer prefers each element be zero.

4.4.3 Optimization Algorithm for Mapping Connectomes of the Brain

The full method consists of two key steps. The first step is to screen the orientations, using GreedyOrientation in Algorithm 3. We then use subgradient descent to optimize the tractography objective using this much-reduced set of parameters. The second step prunes this superset of possible orientations further, often to only a couple of orientations (e.g. $k = 5$ here). The resulting solution only has a small number of active fascicles and orientations for each voxel. We provided a detailed derivation and description of the algorithm in Section 4.4.2.

The optimization given the screened orientations remains convex. The main approximation in the algorithm is introduced from the greedy selection of orientations. We provide approximation guarantees for how effectively the greedy algorithm maximizes the criterion $\bar{g}$. But, this does not characterize whether the criterion itself is a suitable strategy for screening. In the next chapter, we focus our empirical study on the efficacy of this greedy algorithm, which is critical for obtaining efficient solutions for the tractography objective.

Algorithm 5 Brain: Mapping Brain Connectomes

Input: dMRI signal $\mathbf{Y}$, expert three dimensional sparse tensor $\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$, dictionary $\mathbf{D}$, weights (fascicles' contribution) $\mathbf{w}$, voxels vicinity $\mathbf{Vv}$ matrix (group information $\mathcal{G}_{\mathcal{V}}$) and atoms vicinity $\mathbf{Av}$ matrix (group information $\mathcal{G}_{\mathcal{A}}$), maximum iteration *max_iter*, step size *step_size*, termination condition *tolerance*

Ensure: $\|\mathbf{Y}-\underline{\mathbf{\Phi}}\times_1\mathbf{D}\times_3\mathbf{1}\|_F^2+\lambda\sum_{f\in\mathcal{F}}\sum_{\mathcal{G}_{\mathcal{V}}\in\mathcal{V}}\sum_{\mathcal{G}_{\mathcal{A}}\in\mathcal{A}}\|\underline{\mathbf{\Phi}}_{\mathcal{G}_{\mathcal{A}},\mathcal{G}_{\mathcal{V}},f}\|_2$ is minimum

1: **for** $f=1,\ldots,N_f$ **do**
2:     $\underline{\mathbf{\Phi}}_{\boldsymbol{e}}(:,:,f)=\underline{\mathbf{\Phi}}_{\boldsymbol{e}}(:,:,f)*\mathbf{w}(f)$ ▷ Fold $\mathbf{w}$ in $\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$
3: voxels, atoms, fascicles, $\mathbf{Y}\leftarrow$ ComputeCompactData($\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$, $\mathbf{Y}$, fascicles, w)
4: N_a, N_v, $N_f\leftarrow$ size(atoms), size(voxels), size(fascicles)
5: **for** $v=1,\ldots,N_v$ **do**
6:     $\mathbf{aA}\leftarrow$ GreedyOrientation($\mathbf{D}$, k)   ▷ Find indices of active atoms $\mathbf{aA}$ with Algorithm 3
7:     $\mathbf{aF}\leftarrow$ non-zero($\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$, 3)   ▷ Find indices of active fascicles $\mathbf{aF}$ from $\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$
8:     $\underline{\mathbf{\Phi}}(\mathbf{aA},v,\mathbf{aF})\leftarrow Initialization()$ ▷ Initialize $\underline{\mathbf{\Phi}}$ with non-zero values where atoms and fascicles are active
9: $\mathbf{G}_V\leftarrow$ find($\mathbf{Vv}$) ▷ $\mathbf{G}_V(i,j)\in\{0,1\}=1$ if voxel i is in the neighborhood of voxel j
10: $\mathbf{G}_A\leftarrow$ find($\mathbf{Av}$) ▷ $\mathbf{G}_A(i,j)\in\{0,1\}=1$ if atom i is in the neighborhood of atom j
11: **Fmask**, **Amask**, **Fscreen**, **Ascreen** $\leftarrow$ ComputeMasksAndScreens($\underline{\mathbf{\Phi}}$, $\mathbf{G}_V$, $\mathbf{G}_A$)
12: $\mathbf{Y_{diff}}\leftarrow\mathbf{Y}-\underline{\mathbf{\Phi}}\times_1\mathbf{D}\times_3\mathbf{1}$
13: $R(\underline{\mathbf{\Phi}})\leftarrow\sum_{f\in\mathcal{F}}\sum_{\mathcal{G}_{\mathcal{V}}\in\mathcal{V}}\sum_{\mathcal{G}_{\mathcal{A}}\in\mathcal{A}}\|\underline{\mathbf{\Phi}}_{\mathcal{G}_{\mathcal{A}},\mathcal{G}_{\mathcal{V}},f}\|_2$
14: lnew $\leftarrow\|\mathbf{Y_{diff}}\|_F^2+\lambda R(\underline{\mathbf{\Phi}})$
15: niter $\leftarrow 1$
16: **repeat**
17:     lold $\leftarrow$ lnew
18:     g $\leftarrow$ ComputeGradient($\underline{\mathbf{\Phi}}$, $\mathbf{D}$, $\mathbf{Y}$, $\mathbf{G}_V$, $\mathbf{G}_A$, **Fmask**, **Amask**, **Fscreen**, **Ascreen**)
19:     $\underline{\mathbf{\Phi}}\leftarrow\underline{\mathbf{\Phi}}-$ step_size $*g$
20:     $\mathbf{Y_{diff}}\leftarrow\mathbf{Y}-\underline{\mathbf{\Phi}}\times_1\mathbf{D}\times_3\mathbf{1}$
21:     $R(\underline{\mathbf{\Phi}})\leftarrow\sum_{f\in\mathcal{F}}\sum_{\mathcal{G}_{\mathcal{V}}\in\mathcal{V}}\sum_{\mathcal{G}_{\mathcal{A}}\in\mathcal{A}}\|\underline{\mathbf{\Phi}}_{\mathcal{G}_{\mathcal{A}},\mathcal{G}_{\mathcal{V}},f}\|_2$
22:     lnew $\leftarrow\|\mathbf{Y_{diff}}\|_F^2+\lambda R(\underline{\mathbf{\Phi}})$
23:     niter $\leftarrow$ niter $+1$
24: **until** lold $-$ lnew $<$ lold $*$ tolerance $||$ niter $>$ max_iter
25: set small values in $\underline{\mathbf{\Phi}}$ to zero
26: **Output:** $\underline{\mathbf{\Phi}}$

Algorithm 6 ComputeCompactData($\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$, $\mathbf{Y}$, fascicles, w)

Input: expert three dimensional sparse tensor $\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$, the matrix of diffusion signal $\mathbf{Y}$, all fascicles in a tract fascicles, weights (fascicles' contribution) $\mathbf{w}$

1: voxels $\leftarrow$ non-zero($\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$, 2) $\triangleright$ Find necessary voxels from $\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$ (Non-zero elements after summing up the other two dimensions)

2: atoms $\leftarrow$ non-zero($\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$, 1) $\triangleright$ Find necessary atoms(orientations) from $\underline{\mathbf{\Phi}}_{\boldsymbol{e}}$ (Non-zero elements after summing up the other two dimensions)

3: $\mathbf{Y} = \mathbf{Y}(:, \text{voxels})$ $\triangleright$ Remove unnecessary voxels of $\mathbf{Y}$

4: fascicles $\leftarrow$ fascicles(non-zero(w)) $\triangleright$ Remove unnecessary fascicles where contribution (weight) is 0

5: **Output:** voxels, atoms, fascicles, $\mathbf{Y}$

Algorithm 7 ComputeMasksAndScreens($\underline{\mathbf{\Phi}}$, $\mathbf{G}_V$, $\mathbf{G}_A$)

Input: three dimensional sparse tensor $\underline{\mathbf{\Phi}}$, all neighboring voxels $\mathbf{G}_V$, all neighboring orientations $\mathbf{G}_A$

$\mathbf{Emask} \leftarrow (\underline{\mathbf{\Phi}} \times_2 \mathbf{G}_V) \times_1 \mathbf{G}_A$ $\triangleright$ Entry mask tensor

2: $\mathbf{Fmask} \leftarrow \mathbf{Emask} \times_1 \mathbf{1}$ $\triangleright$ Fascicles Mask matrix

$\mathbf{Amask} \leftarrow \mathbf{Emask} \times_3 \mathbf{1}$ $\triangleright$ Atoms Mask matrix

4: $\mathbf{Fscreen} \leftarrow \underline{\mathbf{\Phi}} \times_1 \mathbf{1}$ $\triangleright$ Fascicles Screen matrix. Unlike $\mathbf{Fmask}$, this screen matrix does not contain group information

$\mathbf{Ascreen} \leftarrow \underline{\mathbf{\Phi}} \times_3 \mathbf{1}$ $\triangleright$ Atoms Screen matrix. Unlike $\mathbf{Amask}$, this screen matrix does not contain group information

6: **Output: Fmask, Amask, Fscreen, Ascreen**

Algorithm 8 ComputeGradient($\underline{\mathbf{\Phi}}$, $\mathbf{D}$, $\mathbf{Y}$, $\mathbf{G}_V$, $\mathbf{G}_A$, **Fmask**, **Amask**, **Fscreen**, **Ascreen**)

Input: three dimensional sparse tensor $\underline{\mathbf{\Phi}}$, dictionary $\mathbf{D}$, the matrix of diffusion signal $\mathbf{Y}$, all neighboring voxels $\mathbf{G}_V$, all neighboring orientations $\mathbf{G}_A$, groups of neighboring fascicles **Fmask** and **Fscreen**, groups of neighboring orientations **Ascreen** and **Amask**

grad_p1_x $\leftarrow \mathbf{D}^\top (\underline{\mathbf{\Phi}} \times_1 \mathbf{D} \times_3 \mathbf{1} - \mathbf{Y}) \mathbf{1}^\top$

grad_g1_x1 $\leftarrow |\underline{\mathbf{\Phi}}| \times_1 \mathbf{G}_A^\top$ $\triangleright$ O(number of nonzero elements in $\underline{\mathbf{\Phi}} \times$ number of nonzero elements in $\mathbf{G}_A$)

3: $\underline{\mathbf{A}} \leftarrow \sqrt{\text{grad\_g1\_x1}^2 \times_2 \mathbf{G}_V^\top}$

grad_g1_x3 $\leftarrow \frac{1}{\underline{\mathbf{A}}} \times_2 \mathbf{G}_V$ $\triangleright$ O($N_v \times$ number of nonzero elements in $\underline{\mathbf{A}}$)

grad_g1_x4 $\leftarrow$ grad_g1_x3 $\circ$ grad_g1_x1

6: grad_g1_v $\leftarrow$ (grad_g1_x4 $\times_1 \mathbf{G}_A$) $\circ$ sign($\underline{\mathbf{\Phi}}$)

g $\leftarrow$ grad_p1_x $+ \lambda_g$ grad_g1_v $+ \lambda_1$ sign($\underline{\mathbf{\Phi}}$)

Mask or Screen elements in g with **Fmask**, **Amask** or **Fscreen**, **Ascreen**

9: **Output:** g

Chapter 5

Empirical Results: Reconstructing the Anatomical Structure of Tracts

In this chapter, we present the empirical results of extracting brain connectomes without using any prior tractography models, on several human brains and different tract structures. Our study consists of several parts: acquiring brain diffusion weighted imaging (DWI) data and generating the synthetic datasets since there is no ground truth available for the brain structures invivo, introducing various evaluation metrics to measure the properties of the screening algorithm as well as the proposed objective function and optimization method, developing a visualization algorithm to present the best demonstration of a brain structure or $\underline{\boldsymbol{\Phi}}$, and finally establishing the computational resources used for running the experiments.

5.1   Data Acquisition

The Open Diffusion Data Derivatives (O3D) repository [8] is an open service that allows cognitive and clinical neuroscientists and researchers to have access to new multi-derivative data and integrated processing pipelines. It is generated from three diffusion-weighted Magnetic Resonance Imaging datasets (dMRI) publicly available online called Stanford dataset (STN) [1], Human con-

---

[1]https://purl.stanford.edu/rt034xr8593 and https://purl.stanford.edu/ng782rw8378

nectome project datasets (HCP3T and HCP7T) [2].

The dMRI data from Stanford dataset (STN) contains whole-brain coverage for four subjects, collected with the following configurations: a 3T General Electric Discovery 750 MRI (General Electric Healthcare) using a 32-channel head coil (Nova Medical), and with a dual-spin echo diffusion-weighted sequence, using 96 diffusion-weighting directions and gradient strength of $2,000$ s/mm^2 ($TE = 96.8$ ms). Data spatial resolution was set at 1.5 mm isotropic [8].

The dMRI data of eight subjects from Human connectome project datasets (HCP3T and HCP7T), with whole-brain coverage was collected with Siemens 3T and 7T MRI scanners with $2,000$ s/mm^2 of gradient strength. Data from the 3T and 7T scanners have different properties of resolution. HCP3T has 90 gradient directions and 1.25mm isotropic resolution while HCP7T has 60 gradient directions and 1.05mm isotropic resolution [8].

Three different white matter fascicles tracking methods have been applied on the datasets: tensor-based deterministic tracking, Constrained Spherical Deconvolution (CSD)-based deterministic tracking, and CSD-based probabilistic tracking [8]. In this project, we used tractograms generated by the third method.

This repository consists of brains of 12 subjects, each of which has repeated measures of tractography derivatives for 10 different runs. These datasets have been generated using modern neuroimaging data processing methods such as diffusion-signal modelling, fiber tracking, tractography evaluation, white matter segmentation, and structural connectome construction on different derivatives of diffusion-weighted magnetic resonance imaging (dMRI) with diverse properties of resolution and signal-to-noise ratio [8]. However, as mentioned in Chapter 1, none of these data processing methods guarantees to provide the accurate brain structure of each subject.

---

[2]https://www.humanconnectome.org/study/hcp-young-adult/data-releases

5.1.1 Generating synthetic data

To generate synthetic data for our experiments, we used the dMRI data of four subjects' brains. Then for each of these brains, an expert tractography algorithm was applied to over-generate fascicles, $N_f = 500,000$, as our candidate connectomes to fit the LiFE model [76]. LiFE takes any connectome as input and predicts demeaned diffusion measurements as output in order to evaluate tractography algorithms. We employed LiFE to purify connectomes and prune the number of candidate fascicles. It zeros out the weights of fascicles that do not have significant contribution in reconstructing diffusion signal. This reduces the size of fascicle set about 10–20% for STN, 15–35% for HCP3T, and 20–40% for HCP7T. Also, it decreases the reconstruction error of diffusion signal for all datasets comparing to the reconstruction error of tractography models. We also applied ENCODE to unify encoding of the brain structure and dMRI signal by applying dictionary D.[3]

Using the Automating Fiber-tract Quantification (AFQ) method [110], twenty major human white matter tracts have been segmented:

- Tracts 1 and 2: left and right Anterior Talamic Radiation (ATRl and ATRr),

- Tracts 3 and 4: left and right corticospinal tract (CSTl and CSTr),

- Tracts 5 and 6: left and right Cingulum-Cingulate gyrus (CCgl and CCgr),

- Tracts 7 and 8: left and right Cingulum-Hippocampus portion (CHil and CHir),

- Tracts 9 and 10: Forceps Major (FMJ), and Forceps Minor (FMI),

- Tracts 11 and 12: left and right Inferior Fronto-Occipital Fasciculus (IFOFl and IFOFr),

- Tracts 13 and 14: left and right Inferior Longitudinal Fasciculus (ILFl and ILFr),

---

[3]The dataset can be downloaded from https://brainlife.io

- Tracts 15 and 16: left and right Superior Longitudinal Fasciculus (SLFl and SLFr),

- Tracts 17 and 18: left and right Uncinate Fasciculus (UFl and UFr),

- Tracts 19 and 20: left and right Superior Longitudinal Fasciculus-Temporal part (often referred to as the "arcuate fasciculus", SLFTl and SLFTr).

In this project, we used ENCODE model to generate the predicted signal using three major structures: the Cingulum Hippocampus (Tract 7), the Arcuate Fasciculus (Tract 19), and the ARC-SLF (which is the Arcuate combined with SLFl – which is Tracts 19 and 15).

5.1.2 Datasets

While this dissertation just deals with Arcuate and ARC-SLF of one subject, our long range plan is to investigate the properties of the proposed objective on five different datasets corresponding to four subjects and three major structures in the brain:

- Dataset 1: The first dataset has been generated from the dMRI data of subject 3 in O3D repository, who is one of the four Stanford dataset's subjects. The tract structure we selected for this dataset is Tract 19 aka the Arcuate Fasciculus, hereafter Arcuate. This tract in this dataset has $N_a = 1057$, $N_v = 11823$, $N_f = 868$, and $N_\theta = 96$.

- Dataset 2: The second dataset corresponds to the same subject as Dataset 1 but with another tract, which is the Arcuate combined with one branch of the Superior Longitudinal Fasciculus, SLF1, hereafter ARC-SLF. This tract has $N_a = 1057$, $N_v = 15033$, $N_f = 1100$, and $N_\theta = 96$ in this dataset.

- Dataset 3: The third dataset corresponds to another subject from Stanford Dataset, subject 1, and for Tract 19 aka Arcuate. This dataset has $N_a = 1057$, $N_v = 11488$, $N_f = 508$, and $N_\theta = 96$.

- Dataset 4: The fourth dataset generated from subject 12, who is in HCP7T; this dataset includes the data corresponding to Arcuate aka Tract 7. This dataset has the highest resolution among others with $N_a = 1057$, $N_v = 34910$, $N_f = 1313$, and $N_\theta = 64$.

- Dataset 5: The fifth dataset corresponds to subject 11, from HCP3T for a different tract structure, Tract 7 called Cingulum Hippocampus. We chose this tract to show that our method is able to predict other fascicles' structures rather than Arcuate and ARC-SLF. This dataset has $N_a = 1057$, $N_v = 4542$, $N_f = 210$, and $N_\theta = 64$.

Using these datasets enables comparison to a presumed true underlying $\underline{\mathbf{\Phi}}$. Each dataset has been selected to establish the effectiveness of our method against three main natural variables that might influence the real dMRI data of the brain:

1. Variety of scanning devices and configurations

2. Natural differences in subjects' brains

3. Different structures of tracts

In order to investigate the impact of changing any individual, pair, or triple of these variables, we selected the datasets based on the following reasoning as it is shown in Figure 5.1:

- Dataset 1 and Dataset 2: having the same scanning configuration and subject's brain data, how does our method work on different tracts?

- Dataset 1 and Dataset 3: For similar tract and the same scanning configuration, how does natural differences in subject's brain would affect the output?

- Unfortunately, we did not have access to the data of the same subject's brain being scanned under different devices or configurations. Either or neither of them changes at the same time. Therefore, we cannot
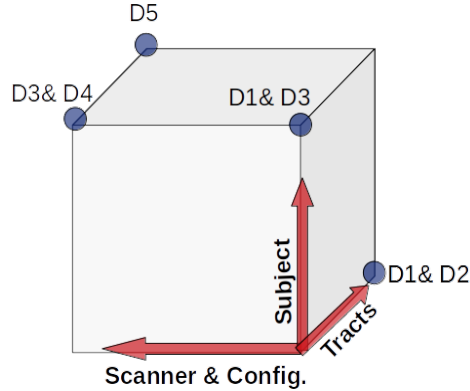
Figure 5.1: Choosing datasets based on different variables

investigate the effect of changing the scanning device and configuration solely on the output results, instead, we design experiments by changing a pair of variables one of them being related to the scanner. Comparing Dataset 3 or 1 with Dataset 4 enables us to learn about the effect of changing both scanner and subject's brain at the same time for a similar tract.

- Finally, the choice of Dataset 5 investigates the performance of proposed method on a different tract scanned with a new configuration from a different brain, meaning that all three variables changing.

Note that, we only provide the empirical results on the first two datasets, Datasets 1 and 2, and the investigation on the other three datasets remain for the future work.

5.2 Inferring Anatomy of Tracts and Evaluating Results

We learn on data generated by an expert connectome solution within the ENCODE model. This allows us to objectively investigate the efficacy of the objective and greedy optimization strategy, because we have access to the ground truth $\underline{\Phi}$ that generated the data. To the best of our knowledge, this is the first completely unsupervised data-driven approach for extracting brain connectomes. We, therefore, focus primarily on understanding the properties

of our optimization approach for tractography. As such, there are no baselines against which to compare our approach. Instead, we define a few metrics to evaluate the performance of our method.

We particularly (a) investigate how effectively our Greedy algorithm selects orientations, (b) investigate how accurately the group regularized objective with this screening approach can reconstruct the diffusion information, and (c) visualize the plausibility of the solutions produced by our method, particularly in terms of smoothness of the fascicles. Even with screening, this optimization when learning over all fascicles and voxels, is prohibitively expensive for running thorough experiments. We therefore focus first on evaluating the model given the assignment of fascicles to voxels, meaning for the following experiments, fascicles are fixed. Because the largest approximation in the algorithm is the greedy selection of orientations, this is the most important step to understand first. For a given set of (greedily chosen) orientations, the objective remains convex with a unique solution. We know, therefore, that further optimizing over fascicles as well would only reduce the reconstruction error; assuming that the optimization criteria is correct.

5.2.1   Screening

We define two error metrics to demonstrate the advantage of GreedyOrientation over OMP for this task. The first is the total number of orientations present in $\underline{\mathbf{\Phi}}$-expert that are not present in $\underline{\mathbf{\Phi}}$ generated by the screening approach, measuring the exactness of the solution. Note that only false negative orientations are critical here since they would be eliminated forever from $\underline{\mathbf{\Phi}}$, which means the optimizer would not be able to find the actual solution, no matter how perfect the objective function is designed. Therefore, only the falsely omitted orientations could impose further error to the optimization stage regarding the predicted $\underline{\mathbf{\Phi}}$. In other words, we expect the weights of the false positive orientations in the orientation candidate set to become zero if the objective function is accurate and the optimizer works well. We also assume that we trust the correctness of synthetic data and consider it as the ground truth.
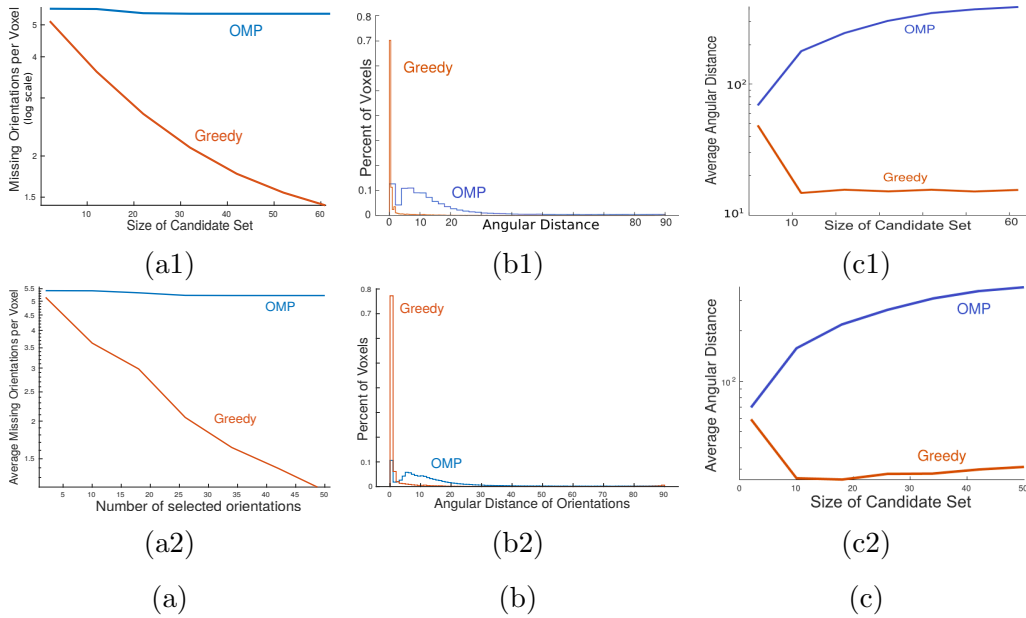
Figure 5.2: The first row corresponds to Arcuate and the second row corresponds to ARC-SLF. **(a):** Average number of missing orientations per voxel in candidate sets of increasing size. **(b):** The distribution of angular distances from the ground truth of OMP and GreedyOrientation after global optimization procedure. The angular distance is the minimum possible distance given some weighted combination of selected orientations calculated based on Algorithm 9. **(c):** Average angular distance between the weighted sum of predicted node orientations and the ground truth in each voxel for candidate sets of increasing size.

The second metric is the minimum possible angular distance between each of the orientations in $\underline{\boldsymbol{\Phi}}$-expert with any arbitrary set of orientations in the corresponding voxel of $\underline{\boldsymbol{\Phi}}$ generated by the screening approach, so that the set would provide the best possible approximation of that orientation. Section 5.3 provides the details of the algorithm measuring the angular distance between two sets of orientations.

We demonstrate the screening method's performance using both error metrics in Figure 5.2. Note that each numerical index shows the corresponding dataset. Figure 5.2a shows the effect of increasing the size of our candidate set of orientations on the number of missing orientations compared to the ground truth. GreedyOrientation's advantage is likely because OMP continually adds dissimilar orientations, thus is less likely to add the exactly correct orientations because these correct orientations are too similar to orientations already

in the candidate set.

Figure 5.2b shows the minimum angular distance given a linear combination of orientations in the candidate set compared to the ground truth. GreedyOrientation has high probability mass near zero, showing that it generates appropriate candidate sets.

Finally, Figure 5.2c shows that the angular distances between the orientations weighted with the optimized weights and ground truth for different size of orientations candidate set. We can clearly see that increasing the size of the orientation set in OMP results in a larger angular distance since more dissimilar orientations are included. On the other hand, the angular distance of candidate sets chosen by GreedyOrientation decreases fast and then stabilized, which indicates that GreedyOrientation forward selection criterion is effectively defined so that the best candidate orientations approximate the ground truth are among the immediate ones. Moreover, according to the graphs corresponding to each dataset, we can infer the minimum best choice of k since a larger value would not affect the final connectome structure significantly. Although, this best choice was $k = 10$, we set $k = 5$ in our experiments, which means that we had larger approximation than the best choice.

We additionally demonstrate the effects of each screening method on final reconstruction error after optimization. Figure 5.3a shows the distribution of reconstruction error over voxels. Starting the optimization with GreedyOrientation leads to much lower bias in the final optimization result than OMP, as demonstrated by the shift of these distributions away from the Ground Truth distribution. In Figure 5.3b, we show the reconstruction error on each step of optimization. The reconstruction error when initialized with orientations generated by OMP is decreasing at a rate several orders of magnitude slower than GreedyOrientation since it has the wrong orientations selected in the candidate set.

5.2.2 Group Sparse Optimization

After $\underline{\mathbf{\Phi}}$ has been initialized with one of the locally greedy screening algorithms, we learn the appropriate weighting of $\underline{\mathbf{\Phi}}$ by optimizing the global objective.
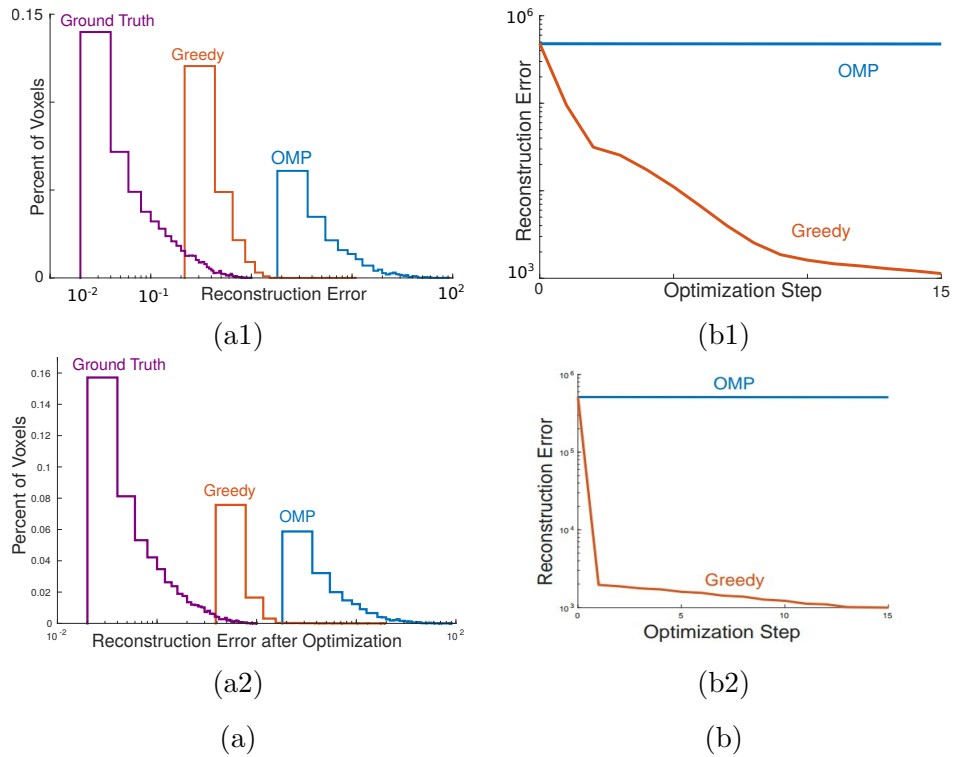
Figure 5.3: The first row corresponds to Arcuate and the second row corresponds to ARC-SLF. **(a):** Comparing the distribution of reconstruction error for ground truth, OMP, and GreedyOrientation over voxels after optimization. **(b):** The improvement of reconstruction error during the steps of gradient decent shows that the objective is not able to improve the OMP selected orientation sets while it is improving the GreedySelection choices constantly.

(a1)   (b1)   (c1)

(a2)   (b2)   (c2)

(a) Ground truth (b) OMP (c) Greedy Orientation
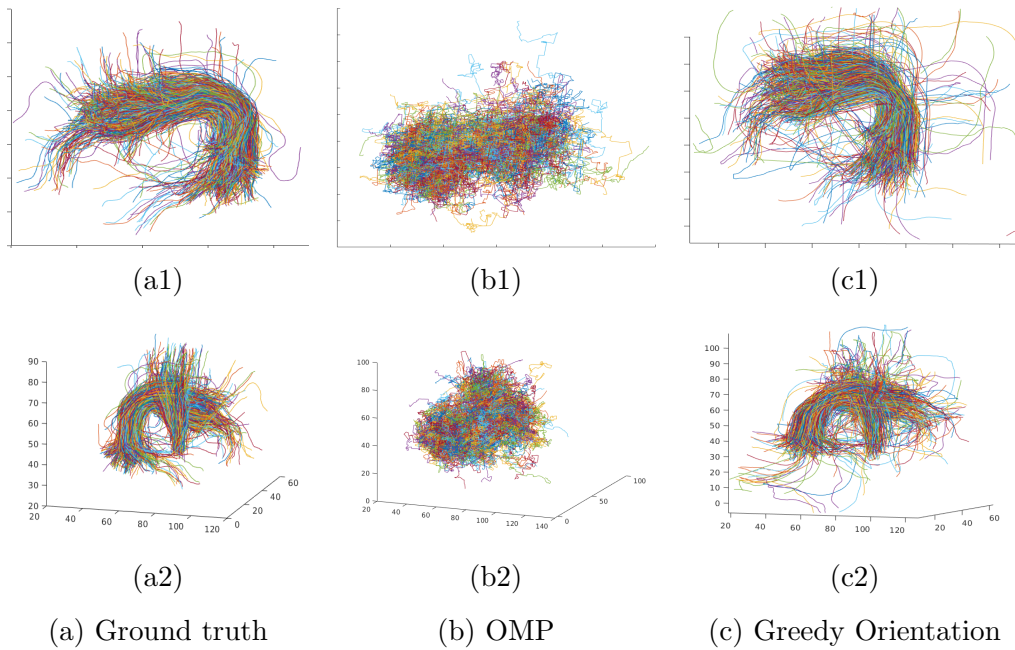
Figure 5.4: Solutions learned after the group sparse optimization for both screening strategies, compared to ground truth. The first row corresponds to Arcuate and the second row corresponds to ARC-SLF. **(a)**: Initializing $\underline{\Phi}$ with expert $\underline{\Phi}$, **(b)**: Initializing $\underline{\Phi}$ with OMP, **(c)**: Initializing $\underline{\Phi}$ with GreedyOrientation.

We applied batch gradient decent with 15 iterations and a dynamic step-size value, which started from 1e-3 and decreased each time the algorithm could not improve the total objective error. The ℓ_1 and group regularizer coefficients were chosen to be 1 for most of the experiments; we tested the following values of the regularization coefficient $[10^{-3}, 10^{-2}, \ldots, 10^2, 10^3]$ and found that results were negligibly affected.

For ℓ_1 regularizer, we applied a proximal operator to truncate weights less than the threshold of 0.001. The derivation of the gradient and optimization procedure have been explained in Sections 4.4.2 and 4.4.3, respectively.

Figure 5.4 visualizes the results of $\underline{\boldsymbol{\Phi}}$ after optimization with both OMP and GreedyOrientation initialization strategies. The visualization algorithm, for a given $\underline{\boldsymbol{\Phi}}$, is given in Section 5.4. Comparing the GreedyOrientation predicted $\underline{\boldsymbol{\Phi}}$ with expert $\underline{\boldsymbol{\Phi}}$ shows that the group regularizer performed well in regenerating macrostructure of all tracts independent of datasets' configuration.

Figure 5.4b shows that the OMP initialization strategy for $\underline{\boldsymbol{\Phi}}$ is not appropriate for this setting, as it prevents the global optimization procedure from generating the desired macrostructure.

To get a better sense for the generated fascicles, Figure 5.5 illustrates the five best and the worst fascicles in terms of the angular distance between the predicted fiber and the corresponding ground-truth one in $\underline{\boldsymbol{\Phi}}$ initialized with GreedyOrientation and OMP. GreedyOrientation produces plausible fascicles in terms of orientation, in some cases seemingly even more so than the ground truth, which was obtained with a tractography algorithm. In the best case, in Figure 5.5a, the reconstruction is highly accurate and the structure and shape of the fascicles clearly align closely with their ground truth counterparts. In the worst case, in Figure 5.5b, GreedyOrientation produces fascicles with sharply changing direction. Looking closer, the worst reconstructed fascicles tend to be long winding fascicles with abrupt direction changes. Because the objective attempts to minimize these features during optimization, these tracts are very difficult to reconstruct. Fascicles such as these are unlikely to occur in the brain, and are likely a result of imperfect tractography methods that were used for creating the ground truth data for this experiment. Solutions with

OMP are generally poor.

5.3   Angular Distance Evaluation Measurement

The goal here is to provide a more precise metric for the angular differences between the nodes in $\underline{\mathbf{\Phi}}$-predict and $\underline{\mathbf{\Phi}}$-expert. It is not a trivial task to measure this metric since a more precise measurement requires finding a one-by-one relationship between the nodes in $\underline{\mathbf{\Phi}}$-predict and $\underline{\mathbf{\Phi}}$-expert. A reasonable way of doing that is to loop over each individual orientation per fascicle-voxel in $\underline{\mathbf{\Phi}}$-expert, $a_{exp} \in \underline{\mathbf{\Phi}}_{exp}(:, v, f)$, and find the optimal solution of active orientations from the candidate set corresponding to the same voxel in $\underline{\mathbf{\Phi}}$-predict so that they could better regenerate the diffusion signal of $\underline{\mathbf{\Phi}}_{exp}(a_{exp}, v, f) \times \mathbf{D}(:, a_{exp})$. Then the angular distance of the vector-sum of activated nodes in $\underline{\mathbf{\Phi}}$-predict with a_{exp} would be calculated and the average angular difference over all a_{exp} per voxel would be reported.

Algorithm 9 Minimum angular distance metric

Input: Two three dimensional sparse tensors $\mathbf{\Phi}_{\text{exp}}$ and $\mathbf{\Phi}_{\text{pred}}$, the expert and predicted brain structures.

1: total_dist $\leftarrow 0$
2: **for** $v_p = 1, \ldots, N_v$ **do**
3:      dist_per_voxel $\leftarrow 0$
4:      ap_set $\leftarrow \mathcal{P}(\{a | a \in \mathbf{\Phi}_{\text{pred}}(:, v_p, :)\})$ ▷ The power set of all orientations active in the current voxel
5:      **for all** $a_{\text{exp}} \in \mathbf{\Phi}_{\text{exp}}(:, v_p, :)$ **do**
6:          $\text{vec}_{\text{exp}} \leftarrow a_{\text{exp}} \times \sum_{f_i \in \mathbf{\Phi}_{\text{exp}}(a_{\text{exp}}, v_p, :)} \mathbf{\Phi}_{\text{exp}}(a_{\text{exp}}, v_p, f_i)$
7:          $min\_dist \leftarrow \infty$
8:          **for all** $s \in$ ap_set **do**
9:              $\text{vec}_{\text{pred}} \leftarrow \sum_{a_i \in s}(a_i \times \sum_{f_i \in \mathbf{\Phi}_{\text{pred}}(a_i, v_p, :)} \mathbf{\Phi}_{\text{pred}}(a_i, v_p, f_i))$
10:             distance $\leftarrow$ Angular_Distance($\text{vec}_{\text{exp}}, \text{vec}_{\text{pred}}$)
11:             **if** $distance < min\_dist$ **then**
12:                min_dist $\leftarrow$ distance
13:          dist_per_voxel $\leftarrow$ dist_per_voxel $+$ min_dist
14:      total_dist $\leftarrow$ total_dist $+$ dist_per_voxel
    Output: total_dist   ▷ The total angular distance

(a1) Best 5 GreedyOrientation (b1) Worst 5 GreedyOrientation

(c1) Best 5 OMP (d1) Worst 5 OMP

(a2) Best 5 GreedyOrientation (b2) Worst 5 GreedyOrientation

(c2) Best 5 OMP (d2) Worst 5 OMP
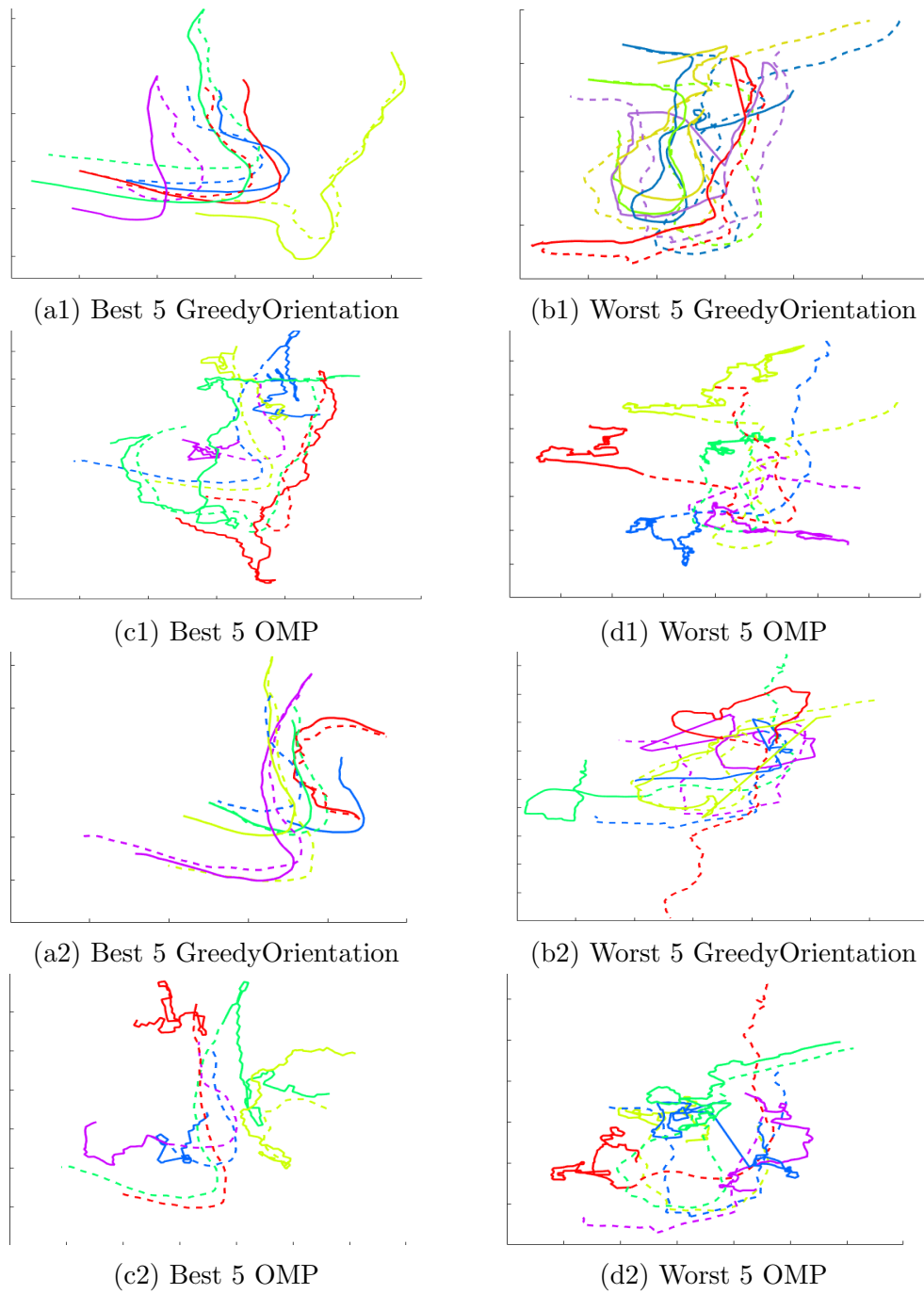
Figure 5.5: Top five best and worst fascicles for OMP and GreedyOrientation after optimization according to reconstruction error. Solid lines show the predicted $\underline{\boldsymbol{\Phi}}$ and dashed lines the ground truth (for the fascicle of the same color). Number 1 corresponds to Arcuate and number 2 corresponds to ARC-SLF.
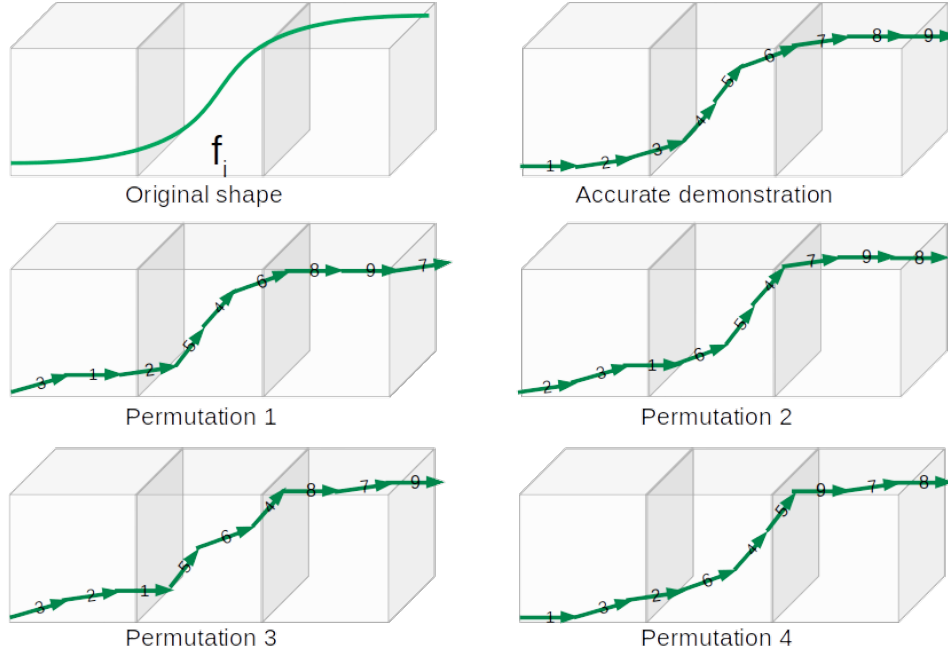
Figure 5.6: A number of different permutations of nodes in the voxels for a fascicle results in a different shape.

5.4 Visualization Algorithm

This section explains the visualization algorithm of the brain connectomes based on the sparse tensor $\underline{\mathbf{\Phi}}$. Each entry of this sparse tensor represents one Node – small chunks of a fascicle chaining together to shape the whole fascicle stream – each having an orientation, being located in a voxel, and belonging to a fascicle. There is no structure in $\underline{\mathbf{\Phi}}$ to indicate how to order the nodes. The nodes do not contain any positional information of the space and this leads to an ambiguity in the accuracy of their order. The only positional information in hand is the coordinates of the voxels. Therefore, displaying an accurate $\underline{\mathbf{\Phi}}$ is itself a challenging problem due to many possible permutations of nodes in a voxel for each fascicle (Figure 5.6).

Our goal is to go over the fascicles one by one and try to plot each at a time. We approach this by selecting one voxel that the fascicle passes through and has the fewest number of neighbouring voxels that also contain the same fascicle. A voxel with these properties should be at one end of the fascicle. Then the algorithm examines all surrounding voxels and chooses the pair of

nodes with the smallest angular distance between them with one in the first voxel and one in the neighbours. We plot the nodes in the first voxel so that the last node in that voxel is the one chosen. Then we move forward through each voxel plotting first the node chosen in the previous pairing followed by the rest of the nodes in the voxel greedily chosen based on angular distance from the last plotted node. Plotting nodes through the neighboring voxels sharing the same fascicle continues by moving on until we reach the other end of the fascicle. Similar to the starting voxel, this ending voxel would have no further neighboring voxel with the same fascicle that has not been met before.

Algorithm 10 Visualize $\mathbf{\Phi}$, the structure of connectomes

---

Input: Any three dimensional sparse tensor $\mathbf{\Phi}$, matrix $\mathbf{A}$ to map the indices of atoms in $\mathbf{\Phi}$ to the Cartesian components of the direction vector of that atom, and matrix $\mathbf{V}$ to map the indices of voxels in $\mathbf{\Phi}$ to the Cartesian coordinates of that voxel

1: **for** $f = 1, \ldots, N_f$ **do**
2:     seen_v $\leftarrow \emptyset$ ▷ Initialize set to keep track of which voxels have been visited for fascicle f up to now
3:     $\mathbf{v} \leftarrow$ get_voxels$(\mathbf{\Phi}, f)$ ▷ Get all voxels that fascicle f passes through
4:     vc $\leftarrow$ GetFirstVoxel(seen_v, $\mathbf{v}$) ▷ Select a voxel that has not been seen for fascicle f
5:     seen_v $\leftarrow$ seen_v $\cup \{$vc$\}$
6:     an $\leftarrow$ PlotFirstVoxelNodes(vc, seen_v, f)   ▷ Plot the nodes of the current voxel and return the next node in the next voxel
7:     **repeat**
8:         vc, ac, seen_v $\leftarrow$ PlotNodes(an, seen_v, f)   ▷ Plot all nodes in the voxel containing an and update vc, ac, and seen_v
9:         anset $\leftarrow$ AllNeighbouringNodes(vc, seen_v)   ▷ Get all active nodes in the neighbouring active to vc
10:        an $\leftarrow$ argmin$_{\text{an}' \in \mathbf{anset}}$(AngularDistance(an$'$, ac))   ▷ Finds the closest node in **anset** to the last plotted node
11:     **until** an is null

We investigate the properties of visualization algorithm for several reasons. First, being able to visualize a valid form of tensor $\underline{\mathbf{\Phi}}$ is a prerequisite for the applicability of our proposed method in the real world for learning brain connectome structure. Trusting the result of this visualization algorithm is critical since we are using it as a tool for validating our optimization method. Furthermore, the best visualization of brain structure is selected among many

---
Algorithm 11 PlotNodes(ac, seen_v, f)

Input: current node to be plotted ac, the set of voxels already visited seen_v, current fascicle f being plotted
 1: vc ← voxel containing ac
 2: seen_v ← seen_v ∪ {vc}
 3: **acset** ← non-zero(Φ(:,f, vc))   ▷ Get all active nodes in the current voxel
 4: **while acset** ≠ ∅ **do**
 5:     **acset** ← **acset** − ac   ▷ Remove ac from **acset**
 6:     Plot(ac)   ▷ Plot the current node
 7:     ac ← argmin$_{ac'∈\mathbf{acset}}$(AngularDistance(ac, ac'))   ▷ Find the next closest node to the previous node in the current voxel
 8: **Output:** vc, ac, seen_v

Algorithm 12 PlotFirstVoxelNodes(vc, seen_v, f)

Input: current voxel vc, the set of voxels already visited seen_v, current fascicle f being plotted
 1: anset ← AllNeighbouringNodes(vc, seen_v)   ▷ Get all active nodes in the neighbouring active to vc
 2: **acset** ← non-zero(Φ(:,f, vc))   ▷ Get all active nodes in the current voxel
 3: ac, an ← argmin$_{an'∈\mathbf{anset},ac'∈\mathbf{acset}}$(AngularDistance(an', ac'))   ▷ Finds the closest nodes between **acset** and **anset**
 4: astack ← empty stack
 5: **while acset** ≠ ∅ **do**
 6:     push ac onto astack
 7:     **acset** ← **acset** − ac   ▷ Remove ac from **acset**
 8:     ac ← argmin$_{ac'∈\mathbf{acset}}$(AngularDistance(ac, ac'))   ▷ Find the next closest node to the previous node in the current voxel
 9: **while** astack not empty **do**
10:     ac ← pop astack
11:     Plot(ac)   ▷ Pop the nodes from astack and plot them
    Output: an   ▷ The closest node in terms of angular distance to the last plotted node
---

different possibilities of $\underline{\Phi}$ representations, and as such, could be considered as an optimization problem. The optimization function could be defined as minimizing the angular distance of the current node and the next node, over the set of possible nodes corresponding to the pair of nearby voxel and the same fascicle. Therefore, we continue this section by talking about visualization performance.

5.4.1 Vetting Visualization Algorithm

We know that a chain of nodes together forms a fascicle and as we mentioned earlier in this section, by encoding fiber structures in $\underline{\Phi}$, we lose the actual positional information of these nodes. However, we still can use the accurate spatial coordinates of the fascicles' nodes in order to investigate the similarity of visualized $\underline{\Phi}$ and the real demonstration of fascicle streamlines. Real demonstration of fascicles is plotted using the precise spatial coordinates of nodes. If both demonstrations were similar, we conclude that visualization algorithm performs well in optimizing the output representation of $\underline{\Phi}$. Otherwise, the dissimilarity might be due to the low quality of encoded brain structure in $\underline{\Phi}$, or because of the deficiencies in the visualization algorithm in choosing the best representation result.

Figure 5.7a shows the plot of fascicle streamlines for subject 11 in O3D repository, run01, and tract07 or Cingulum-Hippocampus. Figure 5.7b demonstrates the same data encoded as tensor $\underline{\Phi}$ and represented by visualization algorithm. The encoded tract structure in $\underline{\Phi}$ has been approximated by using $N_a = 129241$ directions in the space, which means that it has a high resolution in terms of orientations. Therefore, the orientation assigned to each node in $\underline{\Phi}$ is a closer approximation to the actual direction of the same node in the real fiber stream. Comparing these two figures, we can clearly see that they are almost the same, so the ordering of the nodes chosen by visualization algorithm for each fiber is almost similar to the real one.

Now that we can trust the visualization algorithm, we would like to learn about the effect of orientation resolution on the quality of visualization results. This is possible by comparing the demonstration of real fibers, i.e. according

(a)   (b)
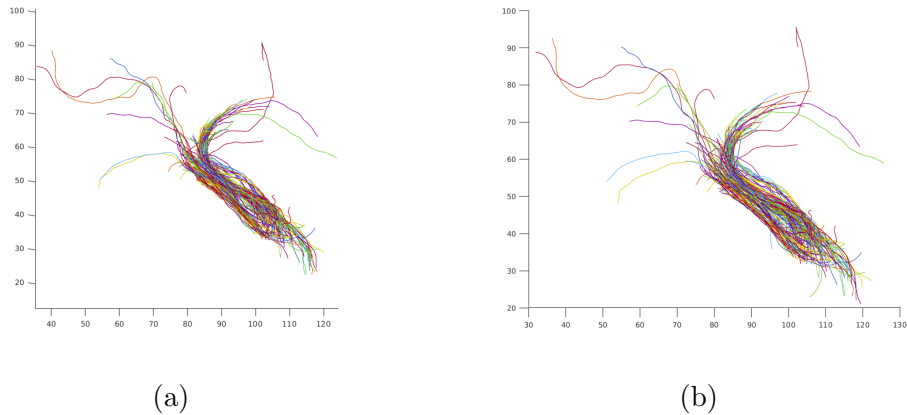
Figure 5.7: **(a):** Demonstration of a tract using positional information of the nodes. **(b):** Visualization of $\underline{\boldsymbol{\Phi}}$ related to the same tact and encoded with relatively large number of orientations.

to the spatial coordinates of nodes, with the visualization of a $\underline{\boldsymbol{\Phi}}$ encoded with smaller number of orientations.

Figure 5.8a also shows the plot of fascicle streamlines for subject 11 in O3D repository, tract07 or Cingulum-Hippocampus, but this time for a different run, e.g. run03. Note that acquiring data of a run for a subject is independent of another run. Therefore, there might be small differences in different runs for the same tract. Figure 5.8b demonstrates visualization result of the same data encoded as tensor $\underline{\boldsymbol{\Phi}}$, but this time the generated LiFE model is built with lower level of space descritization, $N_a = 1057$ orientations. We can see that there are many differences in the details of these two demonstrations, however, the main structure of the tracts are similar.

5.5   Computational Resources

All experiments in this paper were run using an Intel Xeon processor from 2014 with 8 cores at 2.4Ghz each and with 32GB of ram. The code relies heavily on the sparsity of the data, using efficient sparse tensor operations to minimize memory usage and necessary computational resources. Scaling up to larger dimensions or using higher resolution data would greatly increase the total number of entries in the tensors (including empty values), but would

74
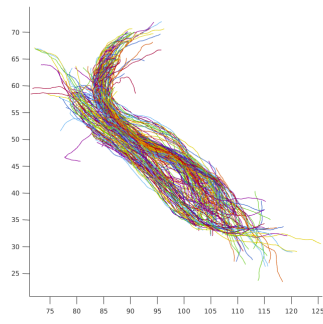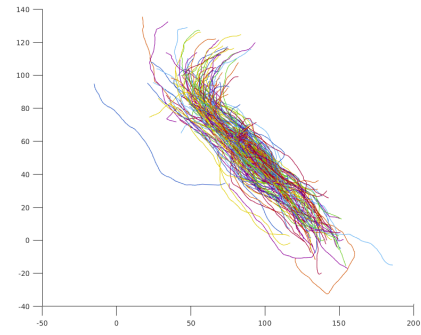
(a)   (b)

Figure 5.8: **(a):** Demonstration of a tract using positional information of the nodes. **(b):** Visualization of $\underline{\mathbf{\Phi}}$ related to the same tact and encoded with relatively smaller number of orientations.

increase the number of *active* entries at a much lower rate.

Chapter 6

Conclusion and Future Work

This chapter summarizes the contributions in the thesis and highlights some future directions for research on this topic.

6.1 Contributions

In this work, we considered the problem of learning macroscopic brain connectomes from dMRI data. This involves inferring locations and orientations of fascicles given local measurements of diffusion of water molecules within the white-matter tissue. This is a problem currently approached by heuristic choices to select preferred recipes of models and parameters among the large set of all potential models and parameters. An initial mapping of the brain connectome and data from the natural brain space into a mutidimensional tensor space allows reformulating the problem of tractography providing a convenient and mathematically tractable framework for machine learning algorithms.

We proposed a new way to formulate this learning problem, using a tensor encoding. Our proposed group sparse objective facilitates the use of optimization algorithms to automatically extract brain structure, without relying on expert tractography solutions. We also proposed an efficient greedy screening algorithm for this objective, and proved approximation guarantees for the algorithm. We finally demonstrated that our specialized screening algorithm resulted in a much better orientations than a generic greedy subselection algorithm, called OMP. We examined our proposed method to learn solutions across brain datasets and multiple anatomies of tracts. The solutions with
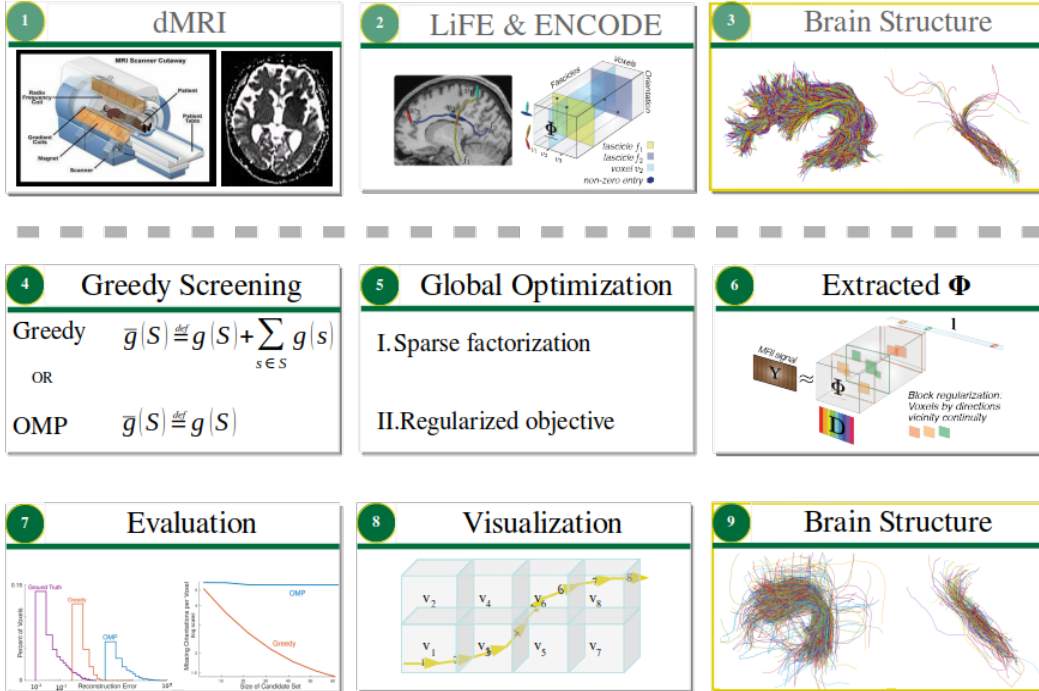
Figure 6.1: A pipeline for extracting brain connectomes. Note that our contributions corresponds to the blocks below the dashed lone.

our group sparse objective, in conjunction with these selected orientations, resulted in smooth fascicles and low reconstruction error of the diffusion data. We also highlighted some failures of the solution, and noted that more needs to be done to get fully plausible solutions.

Figure 6.1 demonstrates the big picture of different stages in this project. The first three stages correspond to the generation of synthetic data. The majority of our contributions starts from stage 3 where the encoded synthetic data is given as the ground truth and the goal is to reconstruct a tract structure that is as similar as possible, in stage 9. Extracting tracts of a connectome from raw dMRI data is a multi-stage pipeline where each stage can introduce different errors and noises. Given the ground truth, our approach might introduce errors in three stages:

1. Screening error: Stage 4 where an approximation of the best orientation choices is made to shrink the dimensionality of the problem. Here, orientations were selected with respect to a proposed gain function.

2. Optimization error: Stage 5 where we try to optimize the tractography objective function with sparse group regularizer over the tensor of brain structure $\underline{\mathbf{\Phi}}$.

3. Visualization error: Stage 8 where visualization algorithm tries to optimize the structural demonstration of each fascicle from $\underline{\mathbf{\Phi}}$ in which the positional information of each node is lost.

In this work, we assumed that the dictionary $\mathbf{D}$ is given by the experts. The ultimate goal, however, is to learn both dictionary $\mathbf{D}$ and brain structure $\underline{\mathbf{\Phi}}$ for any arbitrary input dMRI data – i.e. starting from stage 1, skipping the next two stages, and coming up with the final solution in stage 9. Later, we want to learn dictionary $\mathbf{D}$ and compare it to the expert generated $\mathbf{D}$ to see how well the dictionary has been learned given the synthetic data.

6.2 Future Work

Our tractography learning formulation has the potential to open new avenues for learning-based approaches for obtaining brain connectomes. This preliminary work was necessarily limited, focused on providing a sound formulation and providing an initial empirical investigation into the efficacy of the approximations. The next step is to demonstrate the real utility of a full tractography solution using this formulation. This will involve understanding strengths and weaknesses compared to current tractography approaches; potentially incorporating new regularizers and algorithms; and even incorporating different types of data. We believe that with increased availability of large open datasets and the improvement in data collections technologies, statistical machine learning has the potential to become a more robust and scalable approach to mapping human connectomes [1, 31, 37, 41, 66, 68, 90, 94, 115]. All of this can build on the foundational idea introduced in this work: using a factorization encoding to automatically learn brain structure from data.

Future investigations will be necessary to establish the degree to which the proposed method can be fully applied to multiple, high-resolution data sets.

Moreover, in addition to inferring orientations, we want to be able to predict fascicles as well. The input and output of this process is exactly the same as the current process of optimizing orientations except that, in addition to orientations, we want to optimize fascicles as well instead of assuming that they are fixed. Yet to date, there are multiple methods for group regularization – for example from the image processing literature with similar spatial structure. These include regularizers for total variation [32], general structured regularizers that enable efficient optimization [56] and Laplace regularization and angular and radial regularization [24][1]. All these methods could be explored for this new application. One approach to do this involves extending methods based on group lasso and overlap group lasso regularizers [46], to tensors. Further, regularizers proposed in image analysis for finding planar regions [34], to encourage elongated clusters could also be applied.

In addition, there is a large literature on dictionary learning and multi-way array factorization (tensor factorization) that we could explore. Most of the literature on sparse coding and sparse dictionary learning (see [72] for the seminal paper and [59] for a literature summary) typically considers matrix variables for the sparse coefficients. Even generalizations called tensor sparse coding [89] consider the input to be a multi-way array and the sparse coefficients to be a matrix. Further, many sparse multi-way array factorization algorithms [3, 23, 27, 47] do not consider factorization into a tensor. The closest related algorithm is derived for low-rank regularizers, and for non-negative tensor factorization [108]; though it is not designed for large sparse tensors, it similarly uses a block coordinate descent approach and will likely provide some insight. Though no algorithm has yet to be proposed for our particular optimization setting, the separate advances in factorization algorithms and storing and using large sparse tensors provide a clear opportunity to bring together these advances to develop such algorithms. The developed algorithms for efficient estimation with very large implicit sparse tensors have the potential to fill an important void and impact a wide-range of fields using sparse spatial

---

[1]Though the algorithm in [24] is again for learning the microstructure, the regularization strategies could nonetheless be useful.

data.

All in all, the approach we propose is towards advancing methods for mapping the human connectome. We hope it can make an impact to mapping the connectome to parallel the paradigm shift that has happened in the field of Artificial Intelligence (AI) from rule-based expert systems to data-driven systems and machine learning. This is because the current best practice in the field of tractography is a rule-based process akin to expert systems in AI; where a set of expert rules and models are prescribed following some heuristics. The present work proposes a shift away from human produced tractography methods, toward machine learning, and data driven algorithmic approaches.

References

[1] D. C. Alexander, D. Zikic, A. Ghosh, R. Tanno, V. Wottschel, J. Zhang, E. Kaden, T. B. Dyrby, S. N. Sotiropoulos, H. Zhang, and A. Criminisi, "Image quality transfer and applications in diffusion MRI," *Human Brain Mapping Journal*, pp. 1–65, Mar. 2017. 78

[2] G. Allen, "Sparse higher-order principal components analysis," in *International Conference on Artificial Intelligence and Statistics*, 2012. 52

[3] G. I. Allen, "Sparse higher-order principal components analysis," in *Journal of Machine Learning Research*, Baylor College of Medicine, Houston, United States, Jan. 2012, pp. 27–36. 79

[4] D. Alnaes, T. Kaufmann, G. Richard, E. P. Duff, M. H. Sneve, T. Endestad, J. E. Nordvik, O. A. Andreassen, S. M. Smith, and L. T. Westlye, "Attentional load modulates large-scale functional brain connectivity beyond the core attention networks," *Neuroimage*, vol. 109, pp. 260–272, 2015. 22

[5] F. Aminmansour, A. Patterson, L. Le, Y. Peng, D. Mitchell, F. Pestilli, C. F. Caiafa, R. Greiner, and M. White, "Learning macroscopic brain connectomes via group-sparse factorization," in *Advances in Neural Information Processing Systems*, 2019, pp. 8847–8857. iii

[6] J. S. Anderson, J. A. Nielsen, A. L. Froehlich, M. B. DuBray, T. J. Druzgal, A. N. Cariello, J. R. Cooperrider, B. A. Zielinski, C. Ravichandran, P. T. Fletcher, *et al.*, "Functional connectivity magnetic resonance imaging classification of autism," *Brain*, vol. 134, no. 12, pp. 3742–3754, 2011. 22

[7] Y. Assaf and P. J. Basser, "Composite hindered and restricted model of diffusion (CHARMED) MR imaging of the human brain," *NeuroImage*, vol. 27, no. 1, pp. 48–58, Aug. 2005. 4

[8] P. Avesani, B. McPherson, S. Hayashi, C. F. Caiafa, R. Henschel, E. Garyfallidis, L. Kitchell, D. Bullock, A. Patterson, E. Olivetti, *et al.*, "The open diffusion data derivatives, brain data upcycling via integrated publishing of derivatives and reproducible open cloud services," *Scientific data*, vol. 6, no. 1, p. 69, 2019. 56, 57

81

[9] P. J. Basser, S. Pajevic, C. Pierpaoli, J. Duda, and A. Aldroubi, "In vivo fiber tractography using dt-mri data.," *Magnetic Resonance in Medicine*, vol. 44, no. 4, pp. 625–632, Oct. 2000.   9

[10] P. J. Basser, J. Mattiello, and D. LeBihan, "Estimation of the effective self-diffusion tensor from the nmr spin echo," *Journal of Magnetic Resonance, Series B*, vol. 103, no. 3, pp. 247–254, 1994.   6

[11] D. S. Bassett and O. Sporns, "Network neuroscience," *Nature Neuroscience*, vol. 20, no. 3, pp. 353–364, Feb. 2017.   1

[12] T. E. J. Behrens, M. W. Woolrich, M. Jenkinson, H. Johansen-Berg, R. G. Nunes, S. Clare, P. M. Matthews, J. M. Brady, and S. M. Smith, "Characterization and propagation of uncertainty in diffusion-weighted MR imaging.," *Magnetic resonance in medicine*, 2003.   6, 31

[13] I. Benou and T. R. Raviv, "Deeptract: A probabilistic deep learning framework for white matter fiber tractography," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2019, pp. 626–635.   20

[14] I. Benou and T. Riklin Raviv, "Deeptract: A probabilistic deep learning framework for white matter fiber tractography," in *Medical Image Computing and Computer Assisted Intervention – MICCAI 2019*, D. Shen, T. Liu, T. M. Peters, L. H. Staib, C. Essert, S. Zhou, P.-T. Yap, and A. Khan, Eds., Cham: Springer International Publishing, 2019, pp. 626–635, ISBN: 978-3-030-32248-9.   9

[15] A. Bonnefoy, V. Emiya, L. Ralaivola, and R. Gribonval, "Dynamic Screening: Accelerating First-Order Algorithms for the Lasso and Group-Lasso," *IEEE Transactions on Signal Processing*, 2015.   41, 51

[16] B. G. Booth and G. Hamarneh, "Diffusion mri for brain connectivity mapping and analysis," *MRI: Physics, Image Reconstruction, and Analysis*, pp. 137–171, 2015.   23

[17] C. J. Brown and G. Hamarneh, "Machine learning on human connectome data from mri," *arXiv preprint arXiv:1611.08699*, 2016.   24, 26, 28

[18] C. J. Brown, S. P. Miller, B. G. Booth, K. J. Poskitt, V. Chau, A. R. Synnes, J. G. Zwicker, R. E. Grunau, and G. Hamarneh, "Prediction of motor function in very preterm infants using connectome features and local synthetic instances," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2015, pp. 69–76.   26

[19] G. Brown, A. Pocock, M.-J. Zhao, and M. Luján, "Conditional likelihood maximisation: A unifying framework for information theoretic feature selection," *Journal of machine learning research*, vol. 13, no. Jan, pp. 27–66, 2012.   25

[20] C. F. Caiafa and A. Cichocki, "Computing sparse representations of multidimensional signals using Kronecker bases.," *Neural Computation*, 2013.   52

[21] C. F. Caiafa and F. Pestilli, "Multidimensional encoding of brain connectomes," *Scientific reports*, vol. 7, no. 1, p. 11 491, 2017.   27

[22] C. F. Caiafa, O. Sporns, A. Saykin, and F. Pestilli, "Unified representation of tractography and diffusion-weighted mri data using sparse multidimensional arrays," in *Advances in neural information processing systems*, 2017, pp. 4340–4351.   11, 16

[23] C. Caiafa and A. Cichocki, "Estimation of sparse nonnegative sources from noisy overcomplete mixtures using MAP," *Neural Computation*, vol. 21, no. 12, pp. 3487–3518, 2009.   79

[24] E. Caruyer and R. Deriche, "Diffusion MRI signal reconstruction with continuity constraint and optimal regularization.," *Medical image analysis*, 2012.   9, 79

[25] A. Cichocki, R. Zdunek, A. H. Phan, and S. Amari, *Nonnegative Matrix and Tensor Factorizations: Applications to Exploratory Multi-way Data Analysis and Blind Source Separation*. Wiley, 2009.   51

[26] A. Cichocki, D. P. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. F. Caiafa, and A. H. Phan, "Tensor Decompositions for Signal Processing Applications: From two-way to multiway component analysis.," *IEEE Signal Process. Mag. ()*, 2015.   52

[27] A. Cichocki, D. Mandic, L. De Lathauwer, G. Zhou, Q. Zhao, C. Caiafa, and A. H. Phan, "Tensor decompositions for signal processing applications: from two-way to multiway component analysis," *IEEE Signal Processing Magazine*, vol. 32, pp. 145–163, Mar. 2015.   79

[28] A. Daducci, M. Barakovic, G. Girard, M. Descoteaux, and J.-P. Thiran, "Reducing false positives in tractography with microstructural and anatomical priors," Tech. Rep., 2018.   9

[29] A. Das and D. Kempe, "Submodular meets Spectral: Greedy Algorithms for Subset Selection, Sparse Approximation and Dictionary Selection," in *International Conference on Machine Learning*, 2011.   45, 50

[30] N. U. Dosenbach, B. Nardos, A. L. Cohen, D. A. Fair, J. D. Power, J. A. Church, S. M. Nelson, G. S. Wig, A. C. Vogel, C. N. Lessov-Schlaggar, *et al.*, "Prediction of individual brain maturity using fmri," *Science*, vol. 329, no. 5997, pp. 1358–1361, 2010.   25

[31] A. T. Drysdale, L. Grosenick, J. Downar, K. Dunlop, F. Mansouri, Y. Meng, R. N. Fetcho, B. Zebley, D. J. Oathes, A. Etkin, A. F. Schatzberg, K. Sudheimer, J. Keller, H. S. Mayberg, F. M. Gunning, G. S. Alexopoulos, M. D. Fox, A. Pascual-Leone, H. U. Voss, B. J. Casey, M. J. Dubin, and C. Liston, "Resting-state connectivity biomarkers define neurophysiological subtypes of depression," *Nature Medicine*, pp. 1–16, Dec. 2016.   78

[32] V. Estellers, S. Soatto, and X. Bresson, "Adaptive Regularization With the Structure Tensor.," *IEEE Transactions on Image Processing*, 2015.   79

[33] F. Fei, B. Jie, and D. Zhang, "Frequent and discriminative subnetwork mining for mild cognitive impairment classification," *Brain connectivity*, vol. 4, no. 5, pp. 347–360, 2014.   25

[34] C. Fernandez-Granda and E. J. Candes, "Super-resolution via Transform-Invariant Group-Sparse Regularization.," *ICCV*, 2013.   79

[35] L. R. Frank, "Characterization of anisotropy in high angular resolution diffusion-weighted MRI," *Magnetic Resonance in Medicine*, vol. 47, no. 6, pp. 1083–1099, Jun. 2002.   6

[36] O. Friman, G. Farneback, and C.-F. Westin, "A bayesian approach for stochastic white matter tractography," *IEEE transactions on medical imaging*, vol. 25, no. 8, pp. 965–978, 2006.   23

[37] M. F. Glasser, T. S. Coalson, E. C. Robinson, C. D. Hacker, J. Harwell, E. Yacoub, K. Ugurbil, J. Andersson, C. F. Beckmann, M. Jenkinson, S. M. Smith, and D. C. Van Essen, "A multi-modal parcellation of human cerebral cortex," *Nature Publishing Group*, vol. 536, no. 7615, pp. 171–178, Aug. 2016.   78

[38] I. Guyon, J. Weston, S. Barnhill, and V. Vapnik, "Gene selection for cancer classification using support vector machines," *Machine learning*, vol. 46, no. 1-3, pp. 389–422, 2002.   25

[39] P. Hagmann, O. Sporns, N. Madan, L. Cammoun, R. Pienaar, V. J. Wedeen, R. Meuli, J.-P. Thiran, and P. Grant, "White matter maturation reshapes structural connectivity in the late developing human brain," *Proceedings of the National Academy of Sciences*, vol. 107, no. 44, pp. 19 067–19 072, 2010.   22

[40] M. A. Hall and L. A. Smith, "Feature selection for machine learning: Comparing a correlation-based filter approach to the wrapper.," in *FLAIRS conference*, vol. 1999, 1999, pp. 235–239.   25

[41] H. C. Hazlett, H. Gu, B. C. Munsell, S. H. Kim, M. Styner, J. J. Wolff, J. T. Elison, M. R. Swanson, H. Zhu, K. N. Botteron, D. L. Collins, J. N. Constantino, S. R. Dager, A. M. Estes, A. C. Evans, V. S. Fonov, G. Gerig, P. Kostopoulos, R. C. McKinstry, J. Pandey, S. Paterson, J. R. Pruett, R. T. Schultz, D. W. Shaw, L. Zwaigenbaum, and J. Piven,

"Early brain development in infants at high risk for autism spectrum disorder," *Nature Publishing Group*, vol. 542, no. 7641, pp. 348–351, Feb. 2017.   78

[42] S. M. Hesseltine, Y. Ge, and M. Law, "Applications of diffusion tensor imaging and fiber tractography," *Applied Radiology*, vol. 36, no. 5, p. 8, 2007.   2

[43] M. Heuvel and O. Sporns, "Rich-club organization of the human connectome," *The Journal of neuroscience : the official journal of the Society for Neuroscience*, vol. 31, pp. 15 775–86, Nov. 2011. DOI: 10.1523/ JNEUROSCI.3539-11.2011.   1

[44] A. A. Hofling, J. H. Kim, C. R. Fantz, M. S. Sands, and S.-K. Song, "Diffusion tensor imaging detects axonal injury and demyelination in the spinal cord and cranial nerves of a murine model of globoid cell leukodystrophy," *NMR in Biomedicine: An International Journal Devoted to the Development and Application of Magnetic Resonance In vivo*, vol. 22, no. 10, pp. 1100–1106, 2009.   2

[45] D. Hoza, A. Vlasak, D. Hovrinek, M. Sames, and A. Alfieri, "Dti-mri biomarkers in the search for normal pressure hydrocephalus aetiology: A review," *Neurosurgical review*, vol. 38, no. 2, pp. 239–244, 2015.   2

[46] L. Jacob, G. Obozinski, and J.-P. Vert, "Group lasso with overlap and graph lasso.," *ICML*, 2009.   79

[47] B. Jiang, S. Ma, and S. Zhang, "Tensor principal component analysis via convex optimization," *Mathematical Programming*, vol. 150, no. 2, pp. 423–457, Apr. 2014.   79

[48] B. Jie, D. Zhang, C.-Y. Wee, and D. Shen, "Topological graph kernel on multiple thresholded functional connectivity networks for mild cognitive impairment classification," *Human brain mapping*, vol. 35, no. 7, pp. 2876–2897, 2014.   25

[49] D. Jörgens, Ö. Smedby, and R. Moreno, "Learning a single step of streamline tractography based on neural networks," in *Computational Diffusion MRI*, Springer, 2018, pp. 103–116.   21

[50] K. Kantarci, R. Avula, M. Senjem, A. Samikoglu, B. Zhang, S. Weigand, S. Przybelski, H. Edmonson, P. Vemuri, D. S. Knopman, *et al.*, "Dementia with lewy bodies and alzheimer disease: Neurodegenerative patterns characterized by dti," *Neurology*, vol. 74, no. 22, pp. 1814–1821, 2010.   2

[51] N. C. Keong, A. Pena, S. J. Price, M. Czosnyka, Z. Czosnyka, and J. D. Pickard, "Imaging normal pressure hydrocephalus: Theories, techniques, and challenges," *Neurosurgical focus*, vol. 41, no. 3, E11, 2016.
      2

[52] T. Kolda and B. Bader, "Tensor decompositions and applications," *SIAM Review*, vol. 51, no. 3, pp. 455–500, 2009.   18

[53] T. Konomi, K. Fujiyoshi, K. Hikishima, Y. Komaki, O. Tsuji, H. J. Okano, Y. Toyama, H. Okano, and M. Nakamura, "Conditions for quantitative evaluation of injured spinal cord by in vivo diffusion tensor imaging and tractography: Preclinical longitudinal study in common marmosets," *Neuroimage*, vol. 63, no. 4, pp. 1841–1853, 2012.   4

[54] D. D. Lee and H. S. Seung, "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, no. 6755, p. 788, 1999.   26

[55] T.-W. Lee, "Independent component analysis," in *Independent component analysis*, Springer, 1998, pp. 27–66.   22

[56] S. Lefkimmiatis, A. Roussos, P. Maragos, and M. Unser, "Structure Tensor Total Variation.," *SIAM J. Imaging Sciences*, 2015.   79

[57] X. Li, D. Zhu, X. Jiang, C. Jin, X. Zhang, L. Guo, J. Zhang, X. Hu, L. Li, and T. Liu, "Dynamic functional connectomics signatures for characterization and differentiation of ptsd patients," *Human brain mapping*, vol. 35, no. 4, pp. 1761–1778, 2014.   28

[58] Z. Lin, T. Gong, K. Wang, Z. Li, H. He, Q. Tong, F. Yu, and J. Zhong, "Fast learning of fiber orientation distribution function for mr tractography using convolutional neural network," *Medical physics*, vol. 46, no. 7, pp. 3101–3116, 2019.   21

[59] J. Mairal, F. Bach, J. Ponce, and G. Sapiro, "Online dictionary learning for sparse coding," in *Proceedings of the 26th International Conference On Machine Learning, ICML 2009*, Departement d'Informatique de l'Ecole Normale Superieure, Paris, France, Dec. 2009, pp. 689–696.   79

[60] J. Mitra, K.-k. Shen, S. Ghose, P. Bourgeat, J. Fripp, O. Salvado, K. Pannek, D. J. Taylor, J. L. Mathias, and S. Rose, "Statistical machine learning to identify traumatic brain injury (tbi) from structural disconnections of white matter networks," *NeuroImage*, vol. 129, pp. 247–259, 2016.   22, 23, 25, 26

[61] S. Mori, B. J. Crain, V. P. Chacko, and P. C. M. van Zijl, "Threedimensional tracking of axonal projections in the brain by magnetic resonance imaging," *Annals of neurology*, vol. 45, no. 2, pp. 265–269, Feb. 1999.   7

[62] M. Mørup, L. K. Hansen, and S. M. Arnfred, "Algorithms for Sparse Nonnegative Tucker Decompositions.," *Neural Computation*, 2008.   52

[63] P. Mukherjee, "Diffusion tensor imaging and fiber tractography in acute stroke," *Neuroimaging Clinics*, vol. 15, no. 3, pp. 655–665, 2005.   3

[64] B. C. Munsell, C.-Y. Wee, S. S. Keller, B. Weber, C. Elger, L. A. T. da Silva, T. Nesland, M. Styner, D. Shen, and L. Bonilha, "Evaluation of machine learning algorithms for treatment outcome prediction in patients with epilepsy based on structural connectome data," *Neuroimage*, vol. 118, pp. 219–230, 2015.   26

[65] B. Mwangi, T. S. Tian, and J. C. Soares, "A review of feature reduction techniques in neuroimaging," *Neuroinformatics*, vol. 12, no. 2, pp. 229–244, 2014.   22

[66] G. L. Nedjati-Gilani, T. Schneider, M. G. Hall, N. Cawley, I. Hill, O. Ciccarelli, I. Drobnjak, C. A. M. G. Wheeler-Kingshott, and D. C. Alexander, "Machine learning based compartment models with permeability for white matter microstructure imaging," *Human Brain Mapping Journal*, vol. 150, pp. 119–135, Apr. 2017.   78

[67] P. F. Neher, M. Götz, T. Norajitra, C. Weber, and K. H. Maier-Hein, "A machine learning based approach to fiber tractography using classifier voting," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2015, pp. 45–52.   9

[68] P. F. Neher, M.-A. Cote, J.-C. Houde, M. Descoteaux, and K. H. Maier-Hein, "Fiber tractography using machine learning," *bioRxiv*, pp. 1–20, Jan. 2017.   78

[69] B. Ng, V. Siless, G. Varoquaux, J.-B. Poline, B. Thirion, and R. Abugharbieh, "Connectivity-informed sparse classifiers for fmri brain decoding," in *2012 Second International Workshop on Pattern Recognition in NeuroImaging*, IEEE, 2012, pp. 101–104.   22

[70] T. M. Nir, N. Jahanshad, J. E. Villalon-Reina, A. W. Toga, C. R. Jack, M. W. Weiner, P. M. Thompson, A. D. N. I. (ADNI, *et al.*, "Effectiveness of regional dti measures in distinguishing alzheimer's disease, mci, and normal aging," *NeuroImage: clinical*, vol. 3, pp. 180–195, 2013.   2

[71] M. Ohlson, M. R. Ahmad, and D. von Rosen, "The multilinear normal distribution: Introduction and some basic properties.," *J. Multivariate Analysis ()*, 2013.   31

[72] B. Olshausen and D. Field, "Sparse coding with an overcomplete basis set: A strategy employed by V1?" *Vision research*, vol. 37, no. 23, pp. 3311–3325, 1997.   79

[73] B.-y. Park, J. Seo, J. Yi, and H. Park, "Structural and functional brain connectivity of people with obesity and prediction of body mass index using connectivity," *PloS one*, vol. 10, no. 11, e0141376, 2015.   25

[74] G. J. Parker and D. C. Alexander, "Probabilistic monte carlo based mapping of cerebral connections utilising whole-brain crossing fibre information," in *Biennial International Conference on Information Processing in Medical Imaging*, Springer, 2003, pp. 684–695.   23

[75] Y. C. Pati, R. Rezaiifar, and P. S. Krishnaprasad, "Orthogonal matching pursuit: Recursive function approximation with applications to wavelet decomposition," in *Proceedings of 27th Asilomar conference on signals, systems and computers*, IEEE, 1993, pp. 40–44.   51

[76] F. Pestilli, J. D. Yeatman, A. Rokem, K. N. Kay, and B. A. Wandell, "Evaluation and statistical inference for human connectomes," *Nature Methods*, vol. 11, no. 10, pp. 1058–1063, Sep. 2014.   9, 32, 58

[77] P. Poulin, M.-A. Cote, J.-C. Houde, L. Petit, P. F. Neher, K. H. Maier-Hein, H. Larochelle, and M. Descoteaux, "Learn to track: Deep learning for tractography," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2017, pp. 540–547.   9, 19, 20

[78] G. Prasad, S. H. Joshi, and P. M. Thompson, "Optimizing brain connectivity networks for disease classification using epic," in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, IEEE, 2014, pp. 834–837.   26

[79] D. Pustina, B. Avants, M. Sperling, R. Gorniak, X. He, G. Doucet, P. Barnett, S. Mintzer, A. Sharan, and J. Tracy, "Predicting the laterality of temporal lobe epilepsy from pet, mri, and dti: A multimodal study," *NeuroImage: clinical*, vol. 9, pp. 20–31, 2015.   3

[80] M. Reisert, V. A. Coenen, C. Kaller, K. Egger, and H. Skibbe, "Hamlet: Hierarchical harmonic filters for learning tracts from diffusion mri," *arXiv preprint arXiv:1807.01068*, 2018.   21

[81] J. Richiardi, H. Eryilmaz, S. Schwartz, P. Vuilleumier, and D. Van De Ville, "Decoding brain states from fmri connectivity graphs," *Neuroimage*, vol. 56, no. 2, pp. 616–626, 2011.   25

[82] E. C. Robinson, A. Hammers, A. Ericsson, A. D. Edwards, and D. Rueckert, "Identifying population differences in whole-brain structural networks: A machine learning approach," *NeuroImage*, vol. 50, no. 3, pp. 910–919, 2010.   23, 26

[83] A. Rokem, J. D. Yeatman, F. Pestilli, K. N. Kay, A. Mezer, S. van der Walt, and B. A. Wandell, "Evaluating the accuracy of diffusion MRI models in white matter," *PLoS ONE*, vol. 10, no. 4, e0123272, Apr. 2015.   1

[84] P. Roland and K. Zilles, "Brain atlases-a new research tool," *Trends in neurosciences*, vol. 17, no. 11, pp. 458–467, 1994.   2

[85] R. Rubinstein, M. Zibulevsky, and M. Elad, "Efficient implementation of the K-SVD algorithm using batch orthogonal matching pursuit," Tech. Rep., 2008.   46

[86] T. Schulte, E. M. Müller-Oehring, A. Pfefferbaum, and E. V. Sullivan, "Neurocircuitry of emotion and cognition in alcoholism: Contributions from white matter fiber tractography," *Dialogues in clinical neuroscience*, vol. 12, no. 4, p. 554, 2010. 3

[87] T. B. Sebastian, J. J. Crisco, P. N. Klein, and B. B. Kimia, "Constructing 2d curve atlases," in *Proceedings IEEE Workshop on Mathematical Methods in Biomedical Image Analysis. MMBIA-2000 (Cat. No. PR00737)*, IEEE, 2000, pp. 70–77. 2

[88] Y. Shi and A. W. Toga, "Connectome imaging for mapping human brain pathways," *Molecular psychiatry*, vol. 22, no. 9, pp. 1230–1240, 2017. 8

[89] R. Sivalingam, D. Boley, V. Morellas, and N. Papanikolopoulos, "Tensor sparse coding for region covariances," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, University of Minnesota Twin Cities, Minneapolis, United States, Berlin, Heidelberg: Springer Berlin Heidelberg, Nov. 2010, pp. 722–735. 79

[90] S. M. Smith, T. E. Nichols, D. Vidaurre, A. M. Winkler, T. E. J. Behrens, M. F. Glasser, K. Ugurbil, D. M. Barch, D. C. Van Essen, and K. L. Miller, "A positive-negative mode of population covariation links brain connectivity, demographics and behavior," *Nature Publishing Group*, vol. 18, no. 11, pp. 1565–1567, Sep. 2015. 78

[91] S. Soltani, M. E. Kilmer, and P. C. Hansen, "A Tensor-Based Dictionary Learning Approach to Tomographic Image Reconstruction.," *CoRR abs/1506.04954*, 2015. 52

[92] O. Sporns, "The human connectome: A complex network," *Annals of the New York Academy of Sciences*, vol. 1224, no. 1, pp. 109–125, 2011. 22

[93] G. Swirszcz, N. Abe, and A. C. Lozano, "Grouped Orthogonal Matching Pursuit for Variable Selection and Prediction," *Advances in Neural Information Processing Systems*, 2009. 36

[94] C. M. W. Tax, T. Dela Haije, A. Fuster, C.-F. Westin, M. A. Viergever, L. Florack, and A. Leemans, "Sheet Probability Index (SPI): Characterizing the geometrical organization of the white matter with diffusion MRI," *Human Brain Mapping Journal*, pp. 1–53, Jul. 2016. 78

[95] C. M. Torgerson, A. Irimia, A. D. Leow, G. Bartzokis, T. D. Moody, R. G. Jennings, J. R. Alger, J. D. Van Horn, and L. L. Altshuler, "Dti tractography and white matter fiber tract characteristics in euthymic bipolar i patients and healthy control subjects," *Brain imaging and behavior*, vol. 7, no. 2, pp. 129–139, 2013. 3

[96] J.-D. Tournier, F. Calamante, and A. Connelly, "MRtrix: Diffusion tractography in crossing fiber regions," *International Journal of Imaging Systems and Technology*, vol. 22, no. 1, pp. 53–66, Feb. 2012. 9

[97] J.-D. Tournier, F. Calamante, D. G. Gadian, and A. Connelly, "Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution," *NeuroImage*, vol. 23, no. 3, pp. 1176–1185, Nov. 2004. 9

[98] L. R. Tucker, "Some mathematical notes on three-mode factor analysis.," *Psychometrika*, vol. 31, no. 3, pp. 279–311, Sep. 1966. 27

[99] N. Tzourio-Mazoyer, B. Landeau, D. Papathanassiou, F. Crivello, O. Etard, N. Delcroix, B. Mazoyer, and M. Joliot, "Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain," *Neuroimage*, vol. 15, no. 1, pp. 273–289, 2002. 22

[100] J. L. Ulmer, C. V. Salvan, W. M. Mueller, H. G. Krouwer, G. O. Stroe, A. Aralasmak, and R. W. Prost, "The role of diffusion tensor imaging in establishing the proximity of tumor borders to functional brain systems: Implications for preoperative risk assessments and postoperative outcomes," *Technology in cancer research & treatment*, vol. 3, no. 6, pp. 567–576, 2004. 2

[101] M. P. Van den Heuvel, E. T. Bullmore, and O. Sporns, "Comparative connectomics," *Trends in cognitive sciences*, vol. 20, no. 5, pp. 345–361, 2016. 1

[102] B. A. Wandell, "Clarifying Human White Matter," *Annual Review of Neuroscience*, vol. 39, no. 1, pp. 103–128, Jul. 2016. 1

[103] J. H. Ward Jr, "Hierarchical grouping to optimize an objective function," *Journal of the American statistical association*, vol. 58, no. 301, pp. 236–244, 1963. 22

[104] J. Wasserthal, P. F. Neher, and K. H. Maier-Hein, "Tract orientation mapping for bundle-specific tractography," in *International Conference on Medical Image Computing and Computer-Assisted Intervention*, Springer, 2018, pp. 36–44. 21

[105] V. Wegmayr, G. Giuliari, S. Holdener, and J. Buhmann, "Data-driven fiber tractography with neural networks," in *2018 IEEE 15th International Symposium on Biomedical Imaging (ISBI 2018)*, IEEE, 2018, pp. 1030–1033. 21

[106] J. Weickert, *Anisotropic diffusion in image processing*. Teubner Stuttgart, 1998, vol. 1. 4

[107] Z. J. Xiang, H. Xu, and P. J. Ramadge, "Learning Sparse Representations of High Dimensional Data on Large Scale Dictionaries.," *Advances in Neural Information Processing Systems*, 2011. 41

[108] Y. Xu and W. Yin, "A Block Coordinate Descent Method for Regularized Multiconvex Optimization with Applications to Nonnegative Tensor Factorization and Completion," *SIAM Journal on Imaging Sciences*, vol. 6, no. 3, pp. 1758–1789, Jan. 2013. 79

[109] M. Yang, L. Zhang, X. Feng, and D. Zhang, "Fisher discrimination dictionary learning for sparse representation," in *2011 International Conference on Computer Vision*, IEEE, 2011, pp. 543–550. 28

[110] J. D. Yeatman, R. F. Dougherty, N. J. Myall, B. A. Wandell, and H. M. Feldman, "Tract profiles of white matter properties: Automating fiber-tract quantification," *PloS one*, vol. 7, no. 11, e49790, 2012. 58

[111] B. Yoldemir, B. Ng, and R. Abugharbieh, "Coupled stable overlapping replicator dynamics for multimodal brain subnetwork identification," in *International Conference on Information Processing in Medical Imaging*, Springer, 2015, pp. 770–781. 29

[112] ——, "Stable overlapping replicator dynamics for brain community detection," *IEEE transactions on medical imaging*, vol. 35, no. 2, pp. 529–538, 2015. 29

[113] J. Zeng, P. Zheng, J. Xu, W. Tong, Y. Guo, W. Yang, G. Li, and B. He, "Prediction of motor function by diffusion tensor tractography in patients with basal ganglion haemorrhage," *Archives of medical science: AMS*, vol. 7, no. 2, p. 310, 2011. 3

[114] Z. Zhang and S. Aeron, "Denoising and Completion of 3D Data via Multidimensional Dictionary Learning.," *CoRR abs/1202.6504*, 2015. 52

[115] D. Zhu, N. Jahanshad, B. C. Riedel, and L. Zhan, "Population learning of structural connectivity by white matter encoding and decoding," in *2016 IEEE 13th International Symposium on Biomedical Imaging (ISBI)*, IEEE, 2016, pp. 554–558. 78

[116] D. Zhu, K. Li, L. Guo, X. Jiang, T. Zhang, D. Zhang, H. Chen, F. Deng, C. Faraco, C. Jin, *et al.*, "Dicccol: Dense individualized and common connectivity-based cortical landmarks," *Cerebral cortex*, vol. 23, no. 4, pp. 786–800, 2012. 22

[117] D. Zhu, D. Shen, X. Jiang, and T. Liu, "Connectomics signature for characterizaton of mild cognitive impairment and schizophrenia," in *2014 IEEE 11th International Symposium on Biomedical Imaging (ISBI)*, IEEE, 2014, pp. 325–328. 25

[118] E. Ziv, O. Tymofiyeva, D. Ferriero, A. Barkovich, C. Hess, *et al.*, "A machine learning approach to automated structural network analysis," 2013. 26

[119] S. Zubair and W. Wang, "Tensor dictionary learning with sparse TUCKER decomposition.," *DSP*, 2013. 52