

**UCSF**

**UC San Francisco Electronic Theses and Dissertations**

**Title**

The Modeling of White Matter Architecture and Networks Using Diffusion MRI: Methods, Tools and Applications

**Permalink**

<https://escholarship.org/uc/item/93j4q1hh>

**Author**

Amirbekian, Bagrat

**Publication Date**

2016

Peer reviewed|Thesis/dissertation

Modeling of White Matter Architecture and Networks Using Diffusion MRI:  
Tools, Methods and Applications

by

Bagrat Amirbekian

DISSERTATION

Submitted in partial satisfaction of the requirements for the degree of

DOCTOR OF PHILOSOPHY

in

Bioengineering

in the

GRADUATE DIVISION

of the

UNIVERSITY OF CALIFORNIA, SAN FRANCISCO

AND

UNIVERSITY OF CALIFORNIA, BERKELEY

Copyright 2016  
by  
Bagrat Amirbekian

## **Dedication and Acknowledgements**

I dedicate my dissertation work to my wife, Salem Kimble, and parents Rouben and Sofia Amirbekian. Thank you for the encouragement and support throughout the years. I thank my advisor, Roland Henry, for his guidance and valuable advice, and my friends Anisha Keshavan and Kesshi Jordan for their assistance in editing.

## Abstract

### **The Modeling of White Matter Architecture and Networkings Using Diffusion MRI: Methods, Tools and Applications**

Bagrat Amirbekian

Diffusion magnetic resonance imaging (dMRI) allows us to noninvasively investigate the microstructural properties of brain tissue, and reconstruct the axonal pathways that connect distant brain regions. This enables us to infer the biological processes that give rise to thought and consciousness. However, despite significant advances in both imaging technology and computing power, our ability to estimate connectivity in a single subject using dMRI data remains quite limited. Barriers to accurate single subject estimates include poor accuracy and reproducibility of both fiber tracking and diffusion modeling results, and a difficulty in reproducing the methods of other researchers in this field. As a result, studies using different dMRI methods have drawn conflicting conclusions about the same biological systems. To overcome these barriers, I first present a technique, based on the model residual bootstrap procedure, to estimate the noise in dMRI data and show that this measure is a good indicator of the reproducibility of dMRI measurements. Second, software engineering principles, such as modularization and thorough testing, were implemented and made publicly available in an open source library called Dipy. By providing a single platform where tools and methods from different developers can be implemented using shared constructs and made publicly available to users, Dipy aims to help the community more easily reproduce the findings of other researchers. In the last section of this work, I use these modeling and fiber tracking tools to reconstruct whole brain networks for individual subjects in a large population. The white matter tissue properties projected onto these networks show that regional differences in white matter integrity are strongly associated with body mass index in young, healthy individuals. This association helps explain the reduced cognitive ability in individuals with higher BMI. This study demonstrates the power of using single subject connectivity networks when studying how the human brain and its role in health outcomes. In order to fully unlock the potential of dMRI imaging, methods development needs to continue to focus on improving the reproducibility and accuracy of dMRI studies.

# TABLE OF CONTENTS

Chapter	Page
<b>1 Introduction</b>	<b>1</b>
<b>2 Background</b>	<b>3</b>
2.1 Diffusion MRI Acquisition	3
2.2 Modeling the Diffusion Signal	4
2.2.1 The Diffusion Tensor	4
2.2.2 QBall Imaging and HARDI	6
2.2.3 Constrained Spherical Deconvolution	10
2.2.4 Multi-tissue Constrained Spherical Deconvolution	12
2.3 Fiber Tracking	13
<b>3 Bootstrapping for Noise</b>	<b>15</b>
3.1 Methods	16
3.1.1 Theory	16
3.1.2 Simulation	20
3.1.3 Data collection	21
3.2 Results	22
3.2.1 Simulation Results	22
3.2.2 Human Data	25
3.3 Discussion	28
3.4 Conclusion	30
<b>4 Dipy, a Diffusion Analysis Platform</b>	<b>31</b>
4.1 Model the diffusion signal	31
4.1.1 Model and Fit Classes	32
4.2 Dipy Utilities for Models	33
4.2.1 Gradient Table	33
4.2.2 Spheres and Hemispheres	35
4.3 Local Tracking Components	36
4.3.1 Direction Getter	36
4.3.2 Tissue Classifier	38
4.3.3 Local Tracking Class	39
4.3.4 Multi Voxel Fit Decorator and Class	40

## TABLE OF CONTENTS (cont'd)

4.4	Streamline Utilities . . . . .	41
4.4.1	Connectivity Matrix . . . . .	41
4.4.2	Targeting . . . . .	42
4.4.3	Moving Streamlines . . . . .	42
4.4.4	Density Maps . . . . .	43
<b>5</b>	<b>BMI as it Relates to White Matter Tissues . . . . .</b>	<b>44</b>
5.1	Methods . . . . .	45
5.1.1	Fiber Tracking and Diffusion Metrics . . . . .	45
5.1.2	Statistics . . . . .	46
5.2	Results . . . . .	47
5.3	Discussion . . . . .	49
5.4	Conclusion . . . . .	54
5.5	Appendix . . . . .	55
5.5.1	Effect Map . . . . .	55

## LIST OF FIGURES

	<b>Page</b>
Figure 2.1 The DTI Model.....	5
Figure 2.2 Spherical Harmonic Functions .....	8
Figure 2.3 Multi-Tissue CSD Response Function Axial .....	12
Figure 2.4 Multi-Tissue CSD Response Function Angled.....	13
Figure 2.5 Multi-Tissue Volume Fraction.....	13
Figure 2.6 FACT Fiber Tracking .....	14
Figure 3.1 Noise Estimates Axial.....	23
Figure 3.2 Noise Estimates Coronal .....	24
Figure 3.3 Noise Estimates and RMS Error Simulation .....	25
Figure 3.4 Noise Estimates and RMS Error Human Subject.....	26
Figure 3.5 Noise Estimates Using Different SH Orders .....	28
Figure 4.1 Discrete Hemisphere.....	34
Figure 4.2 Discrete Sphere .....	34
Figure 4.3 Local Tracking.....	36
Figure 5.1 Connectivity Pipeline.....	47
Figure 5.2 Effect Map FA .....	51
Figure 5.3 Effect Map MD .....	52



## LIST OF TABLES

	<b>Page</b>
Table 3.1 Human Data Noise Estimates .....	22
Table 3.2 Simulated Data Noise Estimates.....	26
Table 5.1 Independent Linear Regression Results .....	47
Table 5.2 Multivariate Linear Regression Results .....	48
Table 5.3 PLS Modeling Results .....	49
Table 5.4 Cortical Region Associations with BMI.....	50

# CHAPTER 1

## Introduction

Diffusion weighted magnetic resonance imaging (MRI) is an imaging technique that has allowed unique insights into both the microstructural properties and the organization of cerebral white matter tissues. This technique measures the relative displacement of water molecules in the biological tissues and is highly sensitive to any microstructure which restricts this diffusion. Diffusion MRI is particularly useful in tissues with a high degree of organization, such as the white matter tissues of the brain. This organization means that the restricted diffusion is coherent and measuring the diffusion signal allows us to not only infer the presence or absence of diffusion restricting elements, but also how they're organized and other tissue properties. For example, in the white matter of the brain, the diffusion characteristics of the tissue can give us insight into the organization of axons, the level of myelination of those axons and possible pathology in these tissues. The level of information contained in diffusion imaging makes this imaging technique both powerful and challenging to work with.

Diffusion images cannot generally be read by human specialists, but instead must be modeled using computational techniques. These modeling techniques can produce either composite images or 3d renderings, which can then be used in clinic or for research, or quantitative measurements which can then be used as biomarkers of brain health or disease progression. However, in order to maximize the utility of diffusion imaging one needs to pick the right imaging protocol and modeling approach for a given problem. The modeling of the diffusion signal can be approached both as a local problem, modeling the characteristics of a given brain region or voxel using the diffusion signal specific to that structure. The model can also be thought of as a whole brain model, trying not only to estimate local tissue properties, but also the organization of connections and pathways that contribute to the architecture of the whole brain. Fiber tracking, or tractography, describes the process of using local directional information from brain tissues to build reconstruct these tracts and pathways. These techniques are fairly new and actively being developed and have had

some success in describing the organizational properties of the human brain, or the human connectome.

In this work, I present some common diffusion modeling techniques that have been applied to diffusion MRI, focused specifically on modeling techniques that estimate directional information which can be used for fiber tracking. I present and compare several methods for estimating diffusion MRI noise and in the process discuss how noise estimates for diffusion MRI acquisitions can help identify model failures inform the choice of model for an acquisition type. I further present a framework for thinking about and implement fiber tract reconstruction from diffusion MRI data. Finally I present an application of these methods to a large, public data set for the purpose of understanding the impact of high body mass index on brain health.

## CHAPTER 2

# Background

### 2.1 Diffusion MRI Acquisition

Diffusion weighting in modern neuroimaging applications is most often done using the echo planar imaging (EPI) technique[1, 2]. Because diffusion MRI is attempting to measure the tiny, random motion of water molecules, it is highly sensitive to gross motion of the subject. EPI helps mitigate issues related to subject motion by reducing the time required to acquire each image to less than 100ms. This short acquisition time is also important in diffusion MRI because a complete diffusion MRI data set often includes dozens, or even hundreds, of brain volumes. The time efficiency of the EPI sequence makes it practical to collect complete diffusion MRI data sets in MRI scan sessions that are acceptable for research subjects and patients.

The EPI sequence uses a series of gradient echos, called an echo train, to acquire a large portion of each image slice in a single radio frequency (RF) excitation. Multi-shot EPI sequences have been proposed in order to improve SNR and reduce distortions in diffusion MRI. However, these sequences require longer acquisition times and can produce phase errors in the the data[3]. The result of these limitations has been that single-shot EPI, where the entire image slice is acquired in a single excitation, has become dominant in diffusion MRI acquisitions. The diffusion weighting in a diffusion MRI sequence is introduced by using a pair of diffusion gradients on either side of a 180 degree RF pulse. The first of these diffusion gradients serves to dephase the MRI signal. The following 180 degree pulse inverts the signal so that the second diffusion gradient can now rephase the signal. In the absence of motion, these two gradients have no net effect on the diffusion signal. However, when magnetic particles move during this diffusion weighting procedure, their phases can only partially align, resulting in a signal fallout. Equation 2.1 gives the equation for the diffusion signal,  $S$ , defined in terms of  $S_0$ , the expected signal when no diffusion signal is applied [4]. In this equation,  $\gamma$  is a constant, called the gyromagnetic

ratio,  $G$  is the gradient strength defined in units of Tesla per meter,  $\delta$  is the length, in seconds, of each of two gradient pulses applied, and  $\Delta$  is the time, in seconds, between the first and second diffusion gradient. Finally,  $D(\vec{u})$  is the net diffusivity along the direction of the diffusion gradient,  $\vec{u}$ .

$$S(\vec{u}) = S_0 e^{-\gamma^2 G^2 \delta^2 (\Delta - \frac{1}{3} \delta) D(\vec{u})} \quad (2.1)$$

In practice, all the terms that contribute to the diffusion weighting are grouped into one factor called the b value, where  $b = \gamma^2 G^2 \delta^2 (\Delta - \frac{1}{3} \delta)$ . This b value has the inverse units of diffusivity,  $\frac{s}{m^2}$ , and represents the net diffusion weighting of the sequence. In some applications, like stroke imaging, diffusivity is so altered in the brain tissues that imaging with a few gradient directions is enough to observe and measure the effect [5]. In stroke imaging a metric called apparent diffusion coefficient (ADC) is used. The ADC is the average diffusivity measured along three orthogonal directions, usually the x, y, and z gradient directions. The ADC can be estimated by acquiring three diffusion weighted scans and one scan with no diffusion weighting. However, to do any more complex modeling of the diffusion signal requires acquiring diffusion weighted volumes. The next section discusses different diffusion models, and the types of data need to support the modeling.

## 2.2 Modeling the Diffusion Signal

### 2.2.1 The Diffusion Tensor

Diffusion Tensor Imaging (DTI) is among the oldest and most commonly used models for diffusion MRI applications. In order to fit a diffusion tensor model to the diffusion signal, a minimum of six well dispersed gradient directions are necessary [6]. However, it is often better to acquire at least 20-30 gradient directions to improve the accuracy of the tensor estimate [7]. Even though it may seem that acquiring 30 directions may take significantly more scan time than acquiring 6 directions, in practice a number of excitations (NEX) average is used in DTI studies to improve signal to noise ratio (SNR). Therefore, more gradient directions can be acquired in the same scan time by reducing the NEX value. The increased number of acquired gradient directions compensates for the reduced SNR of each individual image and allows for a more accurate tensor fit. Another consideration when designing an acquisition sequence for DTI is the b-value or values to use. The diffusion tensor best fits data collected at b-values less than  $1500 \frac{s}{m^2}$ , [8]. This following chapter will also discuss the limitations of the tensor model at higher b-values. Some studies have suggested that using multiple b-values can improve the reliability and reproducibility of DTI metrics

[9]. However, DTI studies often use a set of well dispersed diffusion gradients with a constant b-value in addition to one or more non-diffusion weighted images. One advantage of the DTI model is that it can be used to fit most diffusion MRI acquisitions, even sequences optimized for fitting other models tend to produce accurate and reproducible results.

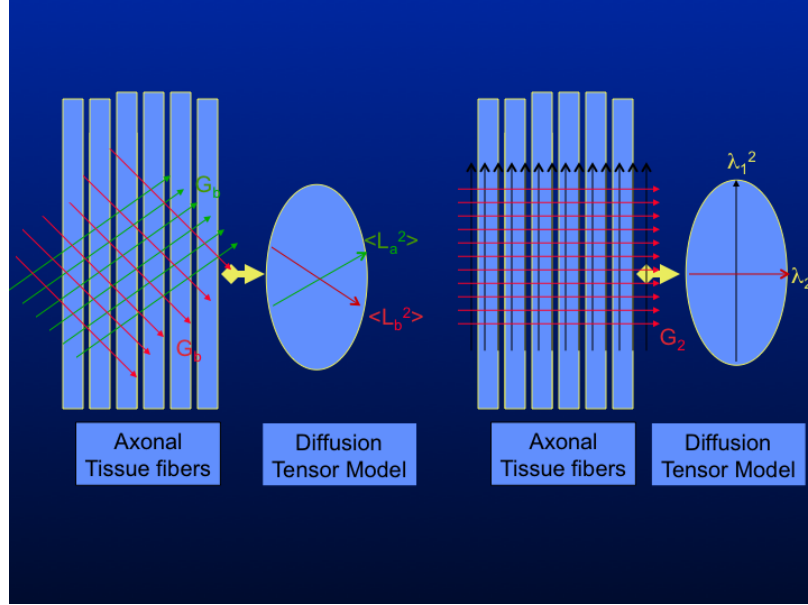


Figure 2.1: The dti model posits that the diffusivity can be estimated as a tensor. The tensor can be represented as an ellipsoid in three dimensional space. The measured diffusivity, when gradient  $G$  is applied, depends on the orientation of the tensor relative to underlying tissues. When the gradient is aligned with the orientation of the tensor, the measured diffusivity is equal to the eigenvalues of the tensor, often called the axial and radial diffusivity.

The diffusion tensor model assumes that the diffusion of water molecules, in the tissues being imaged, can be described by a three dimensional Gaussian distribution. This Gaussian diffusion profile can be described by a 3x3 tensor,  $\mathbf{D}$ . Using this tensor, the diffusivity  $D(\vec{u})$  can be written as  $D(\vec{u}) = \vec{u}^T \mathbf{D} \vec{u}$ . Using this expression for  $D$  and the definition of  $b$  from before, we can express the diffusion signal as such.

$$S(\vec{u}) = S_0 e^{-b(\vec{u}^T \mathbf{D} \vec{u})} \quad (2.2)$$

This equation can be written in a linear form by taking the log of both sides and moving some terms.

$$\log \frac{S(\vec{u})}{S_0} = -b(\vec{u}^T \mathbf{D} \vec{u}) \quad (2.3)$$

The elements of the diffusion tensor can be estimated by imaging with a series of non

colinear gradient directions. Because the tensor is known to be real valued and positive semidefinite, there are 6 unique elements in the tensor. The unique elements of the tensor can be estimated several different ways. Equation 2.3 has a linear form and, when enough signals are available,  $\mathbf{D}$  can be solved for by inverting the system of equations in a way that minimizes the residual error. This approach is known as the linear least squares (LLS) method for estimating diffusion tensors from diffusion MRI signals. However, the LLS method assumes that the error terms, or noise, in the system of equations is independent and identically distributed (iid). It is true that the MRI noise, across gradient directions, can be modeled as iid under some assumptions. However, the linear form of the equation models the log of the signal, so the error terms are in fact not iid at all. To address this issue, the weighted least squares (WLLS) approach is often used to fit estimate the diffusion tensor [10]. While the WLLS approach partially addresses the varying noise distribution, more advanced methods have also been proposed to estimate the diffusion tensor. For example the RESTORE method iteratively fits the diffusion signal, at each iteration identifying outlier signal measurements [11]. While this process allows RESTORE to be robust to patient motion and certain artifacts in the diffusion signal, it comes at the cost of additional computational complexity. The strength of the diffusion tensor comes from the simplicity of its model. This simple model allows us to describe powerful metrics such as mean, axial and radial diffusivity, as well as the seemingly ubiquitous fractional anisotropy. However this strength is also the weakness of this model. The simplicity of the model means that it cannot model the crossing of white matter tracts or multiple water compartments with different diffusivity values. To address these limitations, more complex models have been built to describe the diffusivity profile measured by diffusion MRI.

### **2.2.2 QBall Imaging and HARDI**

High angular resolution diffusion imaging (HARDI) extends the capabilities of DTI by providing more detailed information about the diffusion profile of water molecules than can be described by a tensor model [12]. While there is no strict definition of what qualifies as a HARDI scan protocol, a typical HARDI protocol will consist of many gradient directions, more than 50, acquired at a relatively high b value,  $2000s/mm^2$  or more. The HARDI protocol is better, relative to DTI, and able to capture more complex microstructural tissue architecture not only because of the more complete directional coverage, but also because the higher b-value makes the diffusion MRI scans more sensitive to the diffusion of water molecules at a shorter length scale. HARDI protocols try and maximize angular information relative to scan time, however this can come at the cost of additional information

that might be gained by imaging at multiple b values. Even with that limitation, HARDI scan protocols represent an efficient balance between gathering complete information and minimizing scan times. They are especially useful for fiber tracking applications where directional information is most essential. Data collected using a HARDI scan protocol can be modeled using several methods, including using a diffusion tensor. However, more complex modeling methods such as using a q-ball reconstruction allow us to fully exploit this type of data.

The goal of the q-ball reconstruction is to estimate the diffusion orientation distribution function (ODF) from the diffusion signals. To fully understand the ODF one must first consider the diffusion probability density function (PDF), the density of water molecules as a function of their 3D displacement. This diffusion PDF is the most complete representation of the diffusion profile in an imaging voxel and it turns out that the diffusion MRI signal is actually the 3d fourier transform of the PDF [13, 14]. That is  $S(\vec{q}) = \mathbb{F}[P(\vec{r})]$  where  $\vec{q}$  is defined as  $\vec{q} = \gamma\delta\vec{g}/(2\pi)$  and  $\vec{g}$ ,  $\gamma$ , and  $\delta$  are the diffusion gradient, the gyromagnetic constant and gradient duration [15]. Estimating the PDF directly from the diffusion signal by measuring a full diffusion spectrum in q-space is called diffusion spectrum imaging (DSI). In HARDI imaging however, only one b value is used so only one shell, or "ball", of the diffusion spectrum is collected. This prevents the reconstruction of the full PDF, but the ODF can still be reconstructed from HARDI data.

The diffusion ODF is the radial integral of the diffusion PDF, however this integral has been defined two different ways in the literature. These different definitions of the ODF give slightly different results in the reconstruction, but they both aim to measure the net diffusion of water molecules, as a function of radial direction. The first definition was used by Wedeen et al. when estimating fiber directions using the DSI approach and then later used by Aganj et al. and Tristán-Vega et al. to define a q-ball reconstruction where the radial integral is expressed using a constant solid angle [16, 17]. That is, the ODF is defined as  $\Phi(\vec{u}) = \int P(\vec{r})r^2dr$  where  $\vec{r} = r\vec{u}$  and  $\vec{u}$  is a unit vector. The alternate definition of the ODF,  $\Psi = \int P(\vec{r})dr$  was used by Tuch in the original formulation of the q-ball reconstruction [15]. In both cases, the goal is to represent the directional component of the diffusion profile while collapsing the radial information.

The key insight in q-ball imaging is that the radial integral of the diffusion PDF can be expressed as a great circle integral of the diffusion signal, or some function of the diffusion signal [18, 16]. In reality, this relationship is a simplification, and each of the q-ball variants uses a different set of assumptions in order to reduce what should be a plane integral into a great circle integral, but ultimately expresses the diffusion ODF as a great circle integral over some function of the diffusion signal. Equation 2.4 gives this relationship, known as



the Funk radon transform.

$$F[f(\vec{w})](\vec{u}) = \int \int_{\vec{u}_\perp} f(\vec{w}) \delta(|w| - 1) d^2 \vec{w} \quad (2.4)$$

Under this formulation, both the diffusion signal and the ODF are expressed as functions defined on  $\mathbf{S}^2$  in  $R^3$ .

The spherical harmonic (SH) series represents a set of orthonormal basis functions defined on  $\mathbf{S}^2$ , or the surface of a unit sphere. This series is analogous to the fourier series on the surface in that a fourier decomposition of a function is essentially a frequency representation of the function and any continuous complex function defined on  $\mathbf{S}^2$  can be expressed as a sum over this series. Figure 2.2 shows a visual representation of first several SH functions.

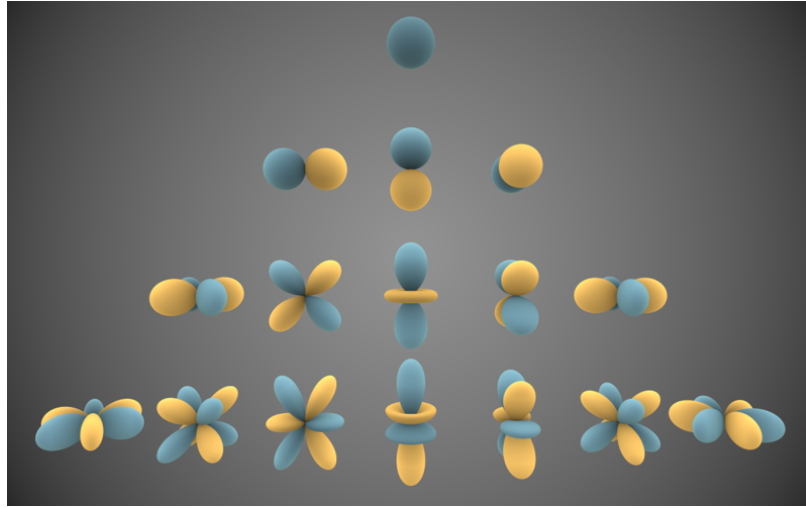


Figure 2.2: A visual representation of the spherical harmonic functions of order 0 to 3, from top to bottom. Notice the antipodal symmetry of the even orders, and the anti-symmetry of the odd orders. [Image](#) by Inigo.quilez is licensed under [CC BY-SA 3.0](#).

This series is particularly useful for q-ball imaging because the Funk radon transform of each SH basis function has an analytical form. Applying the Funk radon transform to any of the SH functions results in a scaled version of the same functions. As a consequence, any function expressed as coefficients to the SH series also has an analytical form. Take for example a function  $f$  defined in terms of the coefficients  $\hat{f}(n, m)$ , where the spherical harmonic of order  $m$  and degree  $n$  is given as  $Y_n^m$ .

$$f(\vec{u}) = \sum_{n=0}^{\infty} \sum_{m=-n}^n \hat{f}(n, m) Y_n^m(\vec{u}) \quad (2.5)$$

The Funk radon transform of this function can be written in terms of the SH functions and the Legendre polynomials,  $P_n$ .

$$F[f(\vec{u})] = \sum_{n=0}^{\infty} P_n(0) \sum_{m=-n}^n \hat{f}(n, m) Y_n^m(\vec{u}) \quad (2.6)$$

With a closed form equation for the Funk radon transform, the ODF of a diffusion MRI signal can be easily computed if the diffusion signal can be estimated as a SH series. In practice, the full SH series is not used to fit the signal because two properties of the diffusion signal allow us to restrict the coefficients. First, because both the signal and the ODF are known to be even, or antipodally symmetric, only the even degrees of the SH series need to be used. That is to say, the coefficients of the odd degree SH functions are known to be zero. Second, both the diffusion signal and ODF are known to be real-valued functions. This constraint allows to one use a set of real values SH functions, which still represent an orthonormal basis over functions defined on the sphere, but which can represent real valued functions only using real valued coefficients. This choice allows the implementation of q-ball imaging without the use of complex numbers, but is mathematically equivalent to using the standard SH functions and enforcing conjugate symmetry in the coefficients. One definition for the real SH functions is given here  $Y_{nm}^R$ .

$$Y_{nm}^R = \begin{cases} \sqrt{2} \cdot \text{Im}[Y_n^m] & \text{if } m > 0 \\ Y_n^0 & \text{if } m = 0 \\ \sqrt{2} \cdot \text{Re}[Y_n^{|m|}] & \text{if } m < 0 \end{cases} \quad (2.7)$$

The SH series must be truncated at a finite degree. It's common to choose the maximum SH degree to be either 6 or 8. The following chapter discusses in more detail some of the considerations that go into picking the maximum SH degree for modeling the diffusion signal. The diffusion signal can now be expressed as  $\vec{e} = Z_g \vec{c}_e$  where  $\vec{c}_e$  is a vector of coefficients,  $Z_g$  is a matrix such that each row represents the truncated SH series evaluated using the directions of the diffusion gradients, and  $\vec{e}$  is some invertible function of the diffusion signals[19]. In order to estimate the diffusion ODF, an inverse of  $Z_g$ , denoted  $Z_g^\dagger$ , must be computed. The Moore-Penrose pseudoinverse works well if the SH degree is truncated at a relatively low value, however inverting  $Z_g$  using a Laplace-Beltrami regularization has been proposed in order to allow higher SH degrees to be used without numerical instability [20]. The ODF, represented as SH coefficients, can be estimated as,  $c_{ODF} = P Z_g^\dagger \vec{e}$ , where  $P$  is the diagonal matrix associated with the Funk radon transform. The diagonal elements of  $P$  consist of the appropriate Legendre polynomial evaluated for

each SH function. This formulation makes q-ball imaging powerful because the ODF can be computed directly from the diffusion signal, without the need for numerical estimation methods.

### 2.2.3 Constrained Spherical Deconvolution

The constrained spherical deconvolution (CSD) model is similar to the q-ball model in that it attempts to estimate directional orientation information using data from a HARDI scan protocol. However, unlike the q-ball model, the CSD model doesn't attempt to estimate the ODF, instead it attempts to model something called the fiber orientation distribution (FOD). In q-ball imaging the water molecule is the basic unit under consideration. Each molecule contributes to the diffusion signal, and from the diffusion signal we attempt to model the net displacement of water molecules, at least in angular terms. The CSD model, however, considers the "fiber" to be the basic imaging unit. If we assume that each fiber contributes equally to the diffusion signal, and that there are no major contributions to the signal from other sources, we can reconstruct the configuration of fibers necessary to produce the observed signal. Biologically a fiber is roughly analogous to a small bundle of axons. If one could isolate such a bundle and image it in isolation, the diffusion signal it would produce would be what's known as the response function. In other words, the response function is the signal produced from a single imaging unit. If the tissue being imaged consists only of these fibers, then the measured signal is simply the sum of the response function from each fiber. The goal of the CSD model is to reconstruct the orientations of the underlying fibers from the diffusion signal.

To build a CSD model, one needs to estimate the response function. Even though one can approximate the response function analytically, for example by assuming that the diffusion PDF associated with each fiber has a gaussian distribution, estimating the response function from data produces the best results in general. The simplest way to do this is to find an area of the an image known to have highly aligned axons, for example a region with high FA values. In this area of the image we can assume that all the fibers share the same orientation. If the fibers are all aligned, then the observed signal is simply a scaling of the response function. The signal from the voxels in this image area should be rotated so the fibers from all the voxels are aligned, we align the fibers with the z axis by convention, then the signals can be averaged to produce an estimate for the response function. This approach works reasonably well, however more advanced methods have also been proposed for estimating the response function of the CSD model. A recursive approach can be used to at once estimate the response function and determine which voxels can be treated as

fully aligned fibers [21]. The idea is to refine the response function until the voxels that contribute to estimating the response converge.

With the response function computed, the CSD method models the signal as a sum over rotated response function. Each fiber in the tissue contributes the same response function to the signal, however the response function associated with each fiber is rotated to match the alignment of the fiber. The diffusion signal is the sum of these rotated response functions. In this formulation the diffusion signal can be expressed as a spherical convolution of the fiber orientations, or the FOD function, and a convolution kernel expressed as rotational harmonics[22]. If both the FOD and the diffusion signal are expressed using SH function, as was the case for q-ball imaging, then the convolution can be expressed as a matrix multiplication,  $\vec{c}_s = R c_{FOD}$ , where  $\vec{c}_s$  and  $c_{FOD}$  are the SH coefficients for the diffusion signal and FOD function respectively and  $R$  is the convolution kernel. Using the CSD model, the full equation for the diffusion signal  $\vec{s}$  becomes:

$$\vec{s} = Z_g R c_{FOD} \quad (2.8)$$

The convolution kernel  $R$  must be estimated from the response function, however if the response function is assumed to be axially symmetric, of equivalently the SH representation of the response function only contains nonzero coefficients for the  $m = 0$  SH functions, then the convolution kernel is known to be diagonal and can be simply estimated by dividing the SH representation of the response function by the SH representation of a delta function [23].

The CSD method uses the constrained deconvolution to estimate the FOD from the diffusion signal. Because the  $R$  matrix is easily invertible, the deconvolution of the diffusion signal can be expressed as a matrix multiplication  $c_{ODF} = R^{-1} \vec{c}_s$ , where  $\vec{c}_s$  is the least squares estimate of the SH representation of the diffusion signal. However, as is often the case with deconvolution problems, this inversion is ill-posed because inverting the  $R$  matrix generally creates large terms which, when applied to noisy diffusion signals, magnify the signal noise. The CSD method proposes to overcome this issue by applying a positivity constraint during the deconvolution. This constraint requires that the FOD be positive on as much of the sphere as is possible. This constraint is biologically motivated because negative fiber densities do not have a meaningful interpretation. The original presentation of the positivity constraint by Tournier et al. proposes an iterative solution of the deconvolution, where the FOD is sampled on a discrete sphere. At each iteration, a regularization term is added to the deconvolution matrix for each negative FOD sample in the previous iteration. The iteration stops when updating the constraints does not reduce the number of negative points on the FOD. This solution does not guarinty a fully positive FOD, but it does allow

a robust and reproducible solution to the deconvolution problem in the presence of noise. Later work has extended the CSD model and proposed a new solution method to the CSD problem that does not require a iterative regularization process.

## 2.2.4 Multi-tissue Constrained Spherical Deconvolution

The multi-tissue constrained deconvolution (MCD) model of diffusion is a natural extension of the CSD diffusion model. Much like the CSD model, the MCD model attempts to reconstruct an FOD from the diffusion signal, however the MCD model allows for a more complex response function. Specifically the MCD approach models the diffusion signal as having contributions from multiple tissue types, each with an independent response function. In order to resolve contributions from multiple response functions, the MTD model needs a multi-shell diffusion acquisition protocol, as opposed to the single-shell HARDI acquisition used by the CSD model. In their initial presentation of this method, Jeurissen et al. use 3 tissue types in the MTD model: white matter, gray matter and cerebrospinal fluid (CSF) [24]. They estimate a response function for each of the 3 tissues, however the gray matter and CSF response functions are assumed to be fully isotropic. A fully isotropic response function has no dependence on gradient direction, only on gradient amplitude. The white matter response function is estimated independently for each q-shell, following the same procedure as the CSD model. Figures 2.3 and 2.4 show the WM response function, computed from the WM tissues of a healthy subject, for a 3-shell, b of 0, 1000, 2000 and 3000, diffusion protocol.

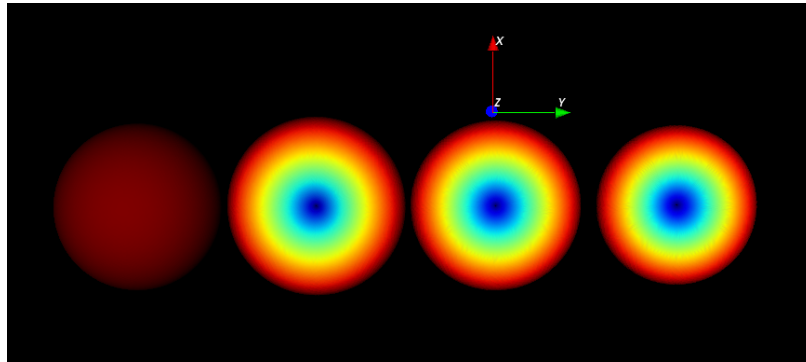


Figure 2.3: The response function of the multi-tissue deconvolution model as viewed from the axial view, camera is pointed along the maximum diffusion direction. From left to right, the b values are 0, 1000, 2000, and 3000  $s/mm^2$ .

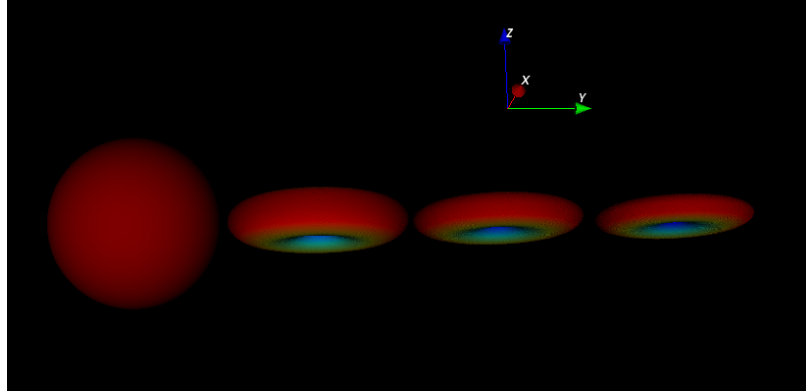


Figure 2.4: The response function of the multi-tissue deconvolution model as viewed from the radial direction, camera is pointed along the restricted diffusion direction. From left to right, the  $b$  values are 0, 1000, 2000, and 3000  $s/mm^2$ .

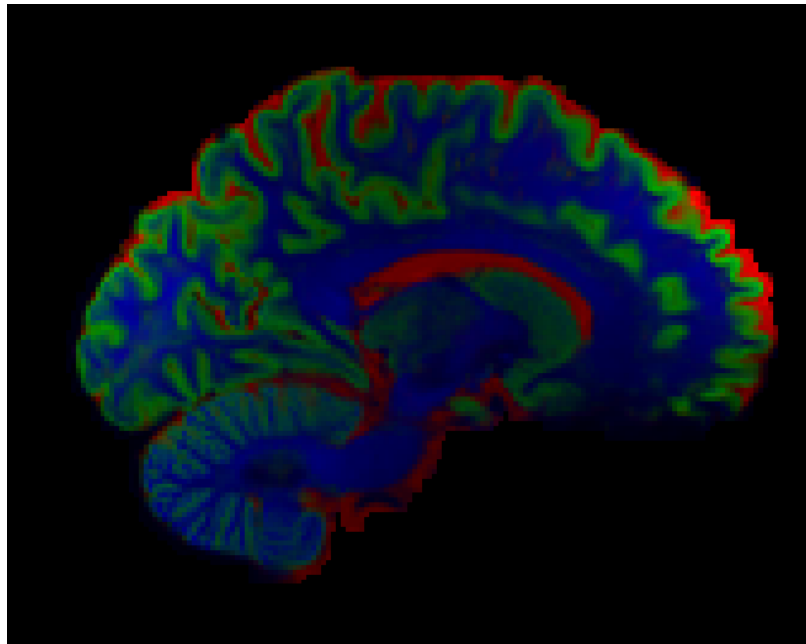


Figure 2.5: A sagittal section of volume fractions estimated using MTD. Red is CSF, Green is gray matter, and blue is white matter. Each voxel is estimated as a mixture of the three.

## 2.3 Fiber Tracking

Each of the diffusion models previously described in this work allow us to model the directional alignment of white matter tissues and extract primary tract directions associated with those tissues. Whether the primary tract directions can be extracted directly, as is the case

when the primary eigen direction of the tensor model is used, or indirectly, as is the case when peak finding is applied to an ODF or FOD, these directions serve as means to follow white matter pathways that connect distant brain regions. An early tracking approach was proposed by Mori et al., and dubbed fiber assignment by continuous tracking or FACT [25]. The basic idea was to follow the path of white matter tissues by tracing along the primary direction estimated using a diffusion model. The FACT algorithm called for the tracing direction to be adjusted at every voxel edge. Figure 2.6 shows an example of such a procedure. More recently, tracking algorithms have generally switched to using a small, fixed step size and reevaluating the tracking direction after each step. Despite small changes, the basic idea hasn't changed all that much.

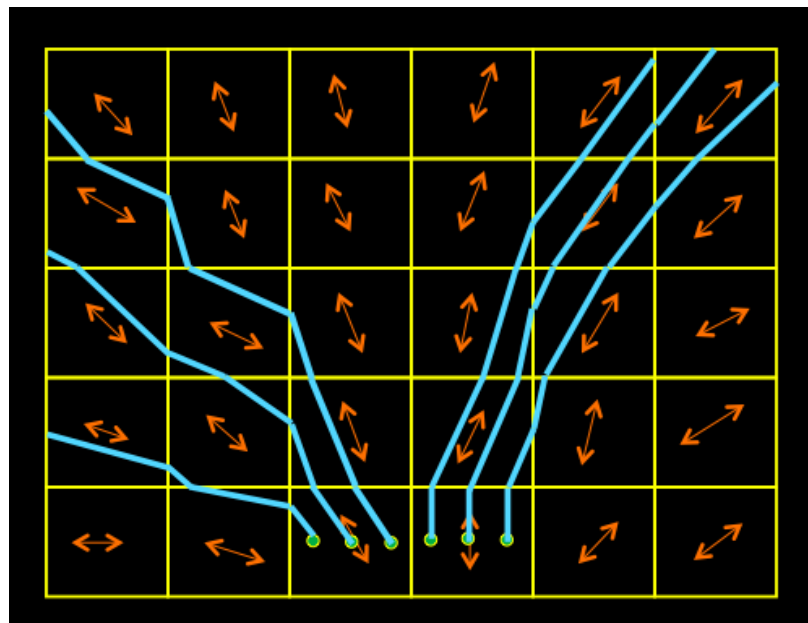


Figure 2.6: A basic diagram of the FACT procedure. Each seed, in green, serves as the starting point for the streamline. The tracking direction is adjusted at each voxel boundary to follow the best estimate of the directional alignment of the local tissue. Tracking continues until a stopping condition is reached.

## CHAPTER 3

# Bootstrapping for Noise

Noise estimates are widely used in image processing tools to evaluate image quality and to estimate signal to noise ratio (SNR). Advanced analysis methods use noise estimates to inform data-driven inferences. The robust model fitting method RESTORE, for example, uses a noise estimate to reject outlier data during the model fitting process. Noise in an MRI image is typically estimated using the signal variance in a region that is expected to have uniform signal. These types of simple noise estimation methods fail in the presence of modern imaging and reconstruction methods. More advanced methods have been proposed to measure noise in images acquired using parallel imaging, but these methods have their own limitations. For example, two recent methods utilize the raw data to estimate the noise [26, 27], but because these methods work with raw data, they must be implemented as part of the vendors' reconstruction algorithm or the raw data must be saved for post hoc analysis. Other noise estimation methods have been proposed which overcome these limitations by analyzing the statistical properties of the intensities in the image background [28, 29]. These background methods rely on the known statistical properties of background voxels, voxels made up primarily of air. These methods simultaneously identify which voxels belong to the background and estimate the noise amplitude. One major limitation of these background methods is that they greatly underestimate the noise if a reconstruction filter is used or if other post processing steps change the statistical distribution of the background voxels[30]. In general, past methods were not developed specifically for dMRI and each of the methods has limitations that must be considered before its use in dMRI applications.

In this work, we present and evaluate several methods for estimating spatially varying noise in high angular resolution diffusion imaging (HARDI) data sets. To our knowledge, this work demonstrates the first method to estimate voxelwise noise maps for dMRI data sets. We focus on HARDI scans because they can be completed in clinically practice acquisitions time and they also support the use of many standard and advanced diffusion models, for example DTI, Q-ball [15] or constrained spherical deconvolution (CSD) [23]. Because the methods we present estimate noise maps directly from signal regions in the



image, they produce detailed noise maps in addition to an overall noise estimate. In this paper, we compare the noise maps produced by each of the four methods to simulated and measured noise maps to evaluate the accuracy and precision of each of the four methods.

## **3.1 Methods**

Each time intensities of an image are measured, a noise sample is drawn from a particular, unknown noise distribution. For the purposes of this study, we attempt to measure the variance of this distribution, termed “noise amplitude”, for each voxel. When an MRI sequence is repeated, the noise amplitude can be estimated directly from these repeated acquisitions. This approach, however, is not practical for either clinical or research applications; if there is extra time in a study for dMRI, the addition of gradient directions or b-values is usually preferred over repeating the same sequence. This study, therefore, focuses on methods that are able to estimate the noise amplitude from a single typical HARDI acquisition. This section describes the four specific methods tested in this study. The noise estimation methods described in this study will be made available as part of Diffusion Imaging in Python (Dipy), an open source software library [31].

### **3.1.1 Theory**

#### **3.1.1.1 B0 noise estimate**

The first method we evaluated is the simplest and does not use a bootstrap procedure. Repeating an entire MRI study is impractical, but it is common practice to acquire multiple minimally diffusion weighted images (b0 images) in a typical dMRI sequence. If multiple b0 images are available, a noise map can be estimated from the observed signal variance across these images. In this paper, we refer to noise amplitude estimated in this manner as the b0 noise estimate.

#### **3.1.1.2 Tensor noise estimate**

The second and third methods we evaluated are both residual bootstrap methods, but employ different models. Residual bootstrap methods have been used to estimate uncertainties in metrics derived from diffusion [32, 33] and to inform fiber tracking [34]. These methods work by generating bootstrap data sets, which are analogous to simulated data acquisitions. These bootstrap data sets can then be used to make inferences about the distribution from

which the data was sampled. In this study, we applied these methods to infer the underlying variance of the diffusion imaging noise distribution.

One model we used for generating bootstrap data sets was the diffusion tensor. Our implementation closely follows the description given by Chung et. al. and is briefly described here. The diffusion signal is modeled as

$$S(g_i) = S_0 e^{-bg_i^T D g_i} \quad (3.1)$$

where  $S(g_i)$  is the diffusion signal measured with gradient  $g_i$ ,  $S_0$  is the signal with no diffusion weighting,  $b$  is the diffusion weighting,  $D$  and is the diffusion tensor. We use this model to generate bootstrap data sets using the following method. The tensor equation can be re-written in the common linear regression form as

$$y = X\beta + \epsilon \quad (3.2)$$

where  $y = [\ln(\frac{S(g_2)}{S_0}), \ln(\frac{S(g_1)}{S_0}), \dots, \ln(\frac{S(g_N)}{S_0})]$  is the log of the normalized measured signals,  $X$  is the design matrix derived from the gradient table, and  $\beta$  is the parameter vector to be fit (composed of the 6 unique elements of the diffusion tensor), and  $\epsilon = [\epsilon_1, \epsilon_2, \dots, \epsilon_N]$  is the set of model residuals. It has been shown that weighted linear least squares (WLLS) outperforms ordinary least squares (OLS) for fitting the diffusion tensor model [35]. The weighted least squares estimate of  $\beta$  is

$$\hat{\beta} = (X^T W X)^{-1} X^T W y \quad (3.3)$$

where  $W$  is a diagonal matrix of weights such that

$$W_{ii} = \hat{S}_i^2 \quad (3.4)$$

$$\hat{S}_i = S_0 e^{\hat{\mu}_i^{OLS}} \quad (3.5)$$

$$\hat{\mu}^{OLS} = H y \quad (3.6)$$

$$H = (X^T X)^{-1} X^T \quad (3.7)$$

The matrix  $H$  is often called the hat matrix. In order to use this fit to generate bootstrap data sets, a set of centered, leverage-corrected residuals must be computed. The raw residuals are given as

$$e = y - X\hat{\beta} \quad (3.8)$$

The centered, leverage-corrected residuals  $q$  are given as

$$r_i = \frac{e_i}{w_i^{-1/2}(1 - h_i)^{1/2}} \quad (3.9)$$

$$q_i = r_i - \bar{r} \quad (3.10)$$

where  $w_i$  and  $h_i$  are the diagonal elements of the  $W$  and  $H$  matrices, respectively. A bootstrap data set,  $y^*$ , is produced by randomly drawing values from  $q$  for  $\epsilon^*$  (with replacement). This is given by

$$y_i^* = x_i\hat{\beta} + w_j^{-1/2}\epsilon^* \quad (3.11)$$

$$S^*(g_i) = S_0 e^{y_i^*} \quad (3.12)$$

Once bootstrap datasets have been generated from the diffusion data set, the noise amplitude at each voxel can be estimated by taking the variance of the bootstrap data sets for that voxel. In this paper, we refer to noise amplitude estimated in this manner as the DTI noise estimate.

### 3.1.1.3 SH noise estimate

The third method for estimating noise amplitudes evaluated in this study is the same as the above method, except the spherical harmonic (SH) functions are substituted for the tensor as the diffusion model. Even though the SH functions are not a diffusion model, when truncated at a maximum SH order, they can be used as a model for residual bootstrap purposes. Because the diffusion signal is real valued and antipodally symmetric we can use the real valued spherical harmonic functions defined in equation 2.7. The signal equation in terms of these SH functions is:

$$y(g_i) = \sum_{n=0}^{\inf} \sum_{m=-n}^l c_{nm} Y_{nm}(g_i) \quad (3.13)$$

where  $c_{nm}$  are the coefficients associated with each of the SH functions  $Y_{nm}$  of degree  $n$  and order  $m$ , and  $g_i$  is a unit vector. It can often be useful to use a single index for the SH functions. We can replace the degree and order with a new single index  $j$  and express

the signal equation as:

$$y(g_i) = \sum_{j=1}^N c_j Y_j(g_i) + e_i \quad (3.14)$$

or

$$y = Xc + e \quad (3.15)$$

where the matrix  $X$  is defined  $X_{ij} = Y_j(g_i)$ , and  $c = \{c_1, c_2, \dots, c_N\}$  is a vector of coefficients. These coefficients can be efficiently estimated using a least squares method and take the form  $c = (X^T X)^{-1} X^T y$ .

The bootstrap samples are created using leverage-corrected and centered residuals. The raw residuals can be computed as

$$e = y - Xc \quad (3.16)$$

$$e = y - X(X^T X)^{-1} X^T y \quad (3.17)$$

$$e = y - Hy = (I - H)y \quad (3.18)$$

where  $H = X(X^T X)^{-1} X^T$  and  $I$  is the identity matrix. The SH fitting is linear, so the leverage correction term can be computed directly from  $H$ . The leverage-corrected residuals are

$$r_i = \frac{e_i}{\sqrt{1 - h_i}} \quad (3.19)$$

where  $h_i = H_{ii}$  are the diagonal entries of the hat matrix. The leverage-corrected residual vector  $r$  can also be written as

$$r = L(I - H)y \quad (3.20)$$

where  $L$  is a diagonal matrix such that  $L_{ii} = 1/\sqrt{1 - h_i}$ . The centered residuals are defined as  $q_i = r_i - \hat{r}$  or equivalently

$$q = C_N L(I - H)y \quad (3.21)$$

where the centering matrix is defined as  $C_N = 1 - \frac{1}{N} \vec{1}\vec{1}^T$ .  $q^*$  is created by sampling from  $q$  with replacement and the bootstrap data sets is defined as

$$S^* = Xc + q^*$$

The noise amplitude of the diffusion signal can be estimated by taking the variance of

these bootstrap samples. We refer to the noise amplitude estimated in this way as the SH noise estimate. In practice, we don't need to generate bootstrap samples  $S^*$  if we notice that  $E[\text{var}(S^*)] = \frac{1}{N-1}\text{var}(q)$ . Also, the matrix  $C_N L(I - H)$  can be computed a-priori, so the computation of the SH noise estimate in each voxel is reduced to two operations, a matrix multiplication and the computation of variance.

#### 3.1.1.4 Non-local means noise estimate

The fourth method we evaluated for estimating noise maps was the application of non-local means filter[36]. This de-noising filter has proven to be a valuable tool for improving data quality of dMRI images before diffusion models are fit to the data or fiber tracking is performed. By taking the difference between the noisy data and the denoised data, we compute a set of residuals, which can be used to generate a voxelwise noise map. Because non-local means filtering requires the average noise amplitude as an input, it needs to be used along with another noise estimation method. For this study, we were only interested in the quality of the noise maps, so we used the known noise amplitude measured from the ground truth data set as the input. This procedure gives an upper bound, best possible result, for the non-local means noise estimation method because in practice the average noise must be computed from the data and may have some error. In order to produce noise maps from the non-local means residuals, the residuals must be scaled so that the average noise amplitude matches the expected value. This step replaces the leverage correction step of the other methods [37]. After scaling, the non-local means noise maps can be compared directly to the ground truth noise maps.

### 3.1.2 Simulation

Model bootstrap methods work well when the model assumptions are satisfied, but tend to overestimate residuals when the model assumptions are violated. In this application, the extent to which model assumptions are violated depends on the b-value, the SH order, and the SNR of the data. In order to understand the performance of the two bootstrap methods as a function of these choices, we used a simulation of two equal-sized fibers crossing at 90 degrees. We simulated each of the two fibers using a tensor model with FA of .7 and mean diffusivity of  $.5 \text{ cm}^2/\text{sec}$ . To simulate Rician noise, we added samples from a standard normal distribution ( $\mu = 0, \sigma = 1$ ) in both the real and complex channels to this simulated signal, and then computed the amplitudes. To vary the SNR, we scaled the signal and held the simulated noise amplitude constant at 1. At each SNR and b-value, we simulated 10,000 independent acquisitions. The simulations were done using the same

b-values and 181-direction gradient table, as were used for acquiring the human subject data. The simulations for this study were performed using the Diffusion Imaging in Python (Dipy) software library [31].

### 3.1.3 Data collection

A human subject was scanned on a 3T Siemens Skyra MRI scanner after obtaining informed consent per the requirements of our local institutional review board. The human subject was scanned to acquire 6 total data sets: a repetition data set and a HARDI data set at each b-value ( $b=1000, 2000, \text{ and } 3000 \frac{s}{mm^2}$ ). For each b-value, the repetition data set consisted of 3 gradient directions acquired 80 times each, along with 7 images with no diffusion weighting. Each HARDI data set consisted of 181 non-collinear gradient directions and 7 images with no diffusion weighting. All images were acquired at 2.2mm isotropic resolution,  $TR = 3s$ , using a multiband factor of 2.

We used a motion and eddy current correction scheme similar to the method employed by FSL’s “`eddy_correct`”. This correction scheme registers all the diffusion-weighted images to a reference to compensate for scaling and shearing introduced by eddy and subject motion during the scan. In order to compare voxelwise between the HARDI and repetition data sets, we combined the  $b_0$  images from the two data sets to make one reference image. This method produced one eddy correction reference for each of the 3 b-values acquired. To produce this reference, we first averaged all  $b_0$  images from the two data sets. We then registered those same  $b_0$  images to the average image, and computed the average transform to ensure that all images are moved as little as possible. We then used this image as the reference for our motion and eddy current correction. The end result of this process was a repetition and a HARDI data set that were eddy-corrected and co-registered to each other. The repetition data set was made up of repeated signal measurements along three gradient directions. By taking the variance of the repeated measurements at each voxel and averaging the three results, we created the “ground truth noise maps”. Noise maps were also estimated from the HARDI data sets using each of the candidate methods described. Even though we do not refer to the ground truth noise maps as estimates for brevity and clarity, they are in fact estimates as well and when comparing the candidate noise maps to the ground truth noise maps, one must account for the standard error in both noise maps. For our quantitative comparisons of the different noise estimates to the ground truth noise maps, we limited the analysis to a brain tissue mask. The brain tissue mask was created by first fitting a tensor model to each of the data sets and taking the voxels with fractional anisotropy (FA) greater than .1 and axial diffusivity less than  $2 \frac{cm^2}{s}$ . We then performed

Table 3.1: The median DTI, SH (order 6), and B0 noise estimates and RMS errors computed from a single subject. The data was normalized using the repetition ground truth so that the expected noise amplitude is 1.

b-value ( $\frac{s}{mm^2}$ )	1000	2000	3000
DTI noise estimate	1.08	1.11	1.31
DTI RMS error	0.44	0.31	0.49
SH-6 noise estimate	1.01	0.96	1.13
SH-6 RMS error	0.23	0.22	0.19
B0 noise estimate	1.00	0.97	0.85
B0 RMS error	1.44	1.26	0.91
NLM RMS error	1.08	0.84	0.51
Median B0 SNR	24.92	19.68	18.02
Median DWI SNR	11.79	5.67	4.02

binary erosion to the result to produce the final brain tissue mask.

## 3.2 Results

Overall, the SH noise estimates and noise maps were the most accurate and most precise. Figure 3.1 and 3.2 show a qualitative comparison of the noise maps produced by the 4 single-acquisition methods along side the repetition ground truth for human data at b-value of 3000. It is clear from visual comparison of the four noise maps that the SH noise map (C) most closely matches the ground truth (A). The DTI noise map (B) captures some of the features present in the repetition ground truth, however it overestimates the noise amplitude in several regions. The B0 noise map (D) is too grainy to capture any of the fine features in the ground truth noise map. The B0 noise estimation method may still be useful as a global estimate of the average noise amplitude in an image, but it is not useful as a noise map. The noise maps produced using the non-local means filtering (E) also had a tendency to overestimate the noise specific tissue regions. In the next section, we present a quantitative analysis of accuracy and precision of the 4 single-acquisition noise estimation methods.

### 3.2.1 Simulation Results

In simulation, the SH noise estimates show near-perfect behavior, while the DTI noise estimates are positively biased. Table 3.2 shows the average noise estimates and root mean square (RMS) errors obtained from the simulated data using the two residual bootstrap methods. The average noise estimate is a measure of the method accuracy. The RMS error

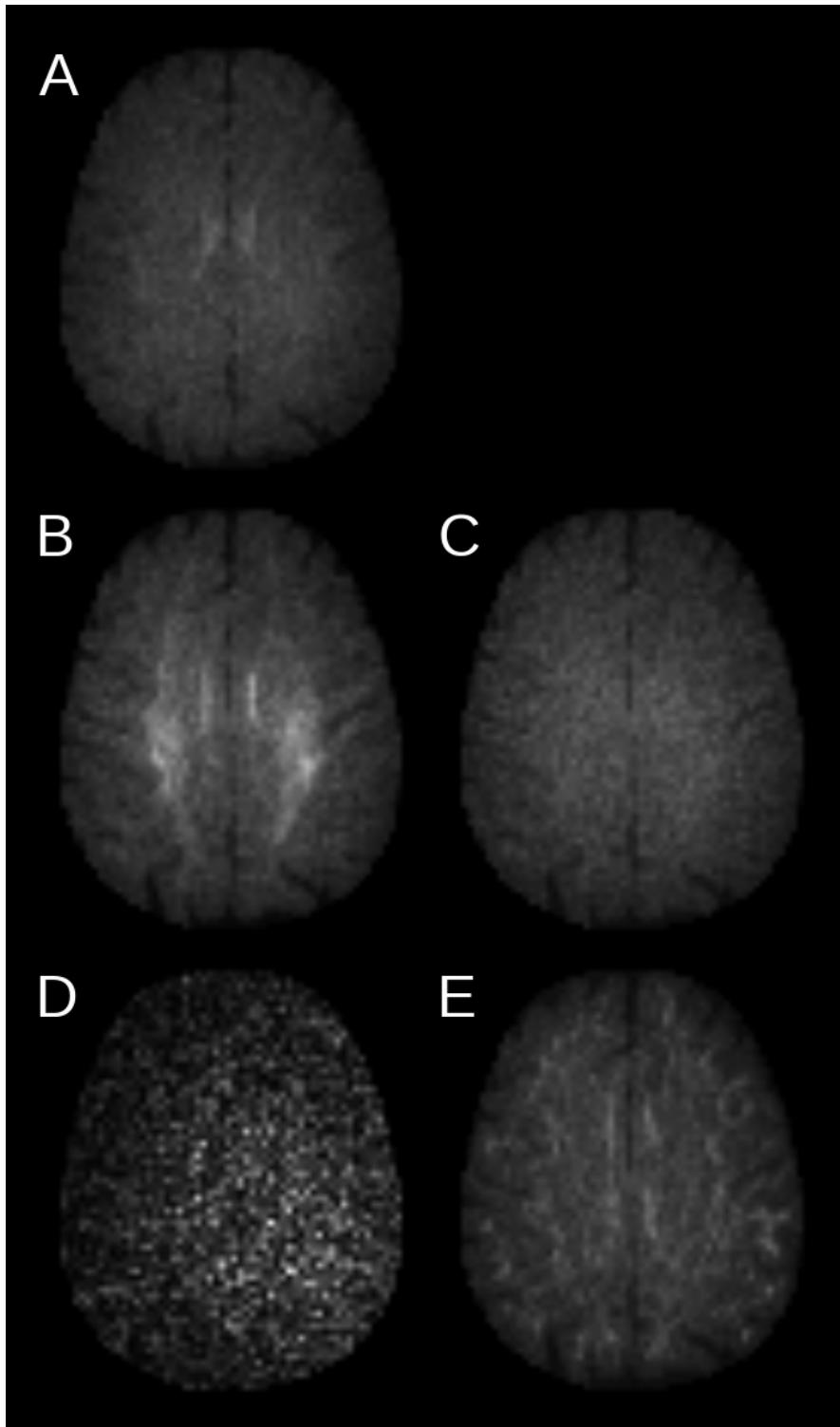


Figure 3.1: The ground truth (A), DTI estimate (B), SH estimate (C), B0 estimate (D) and non-local means estimate(E) noise maps produced from the  $b=3000$  data set.



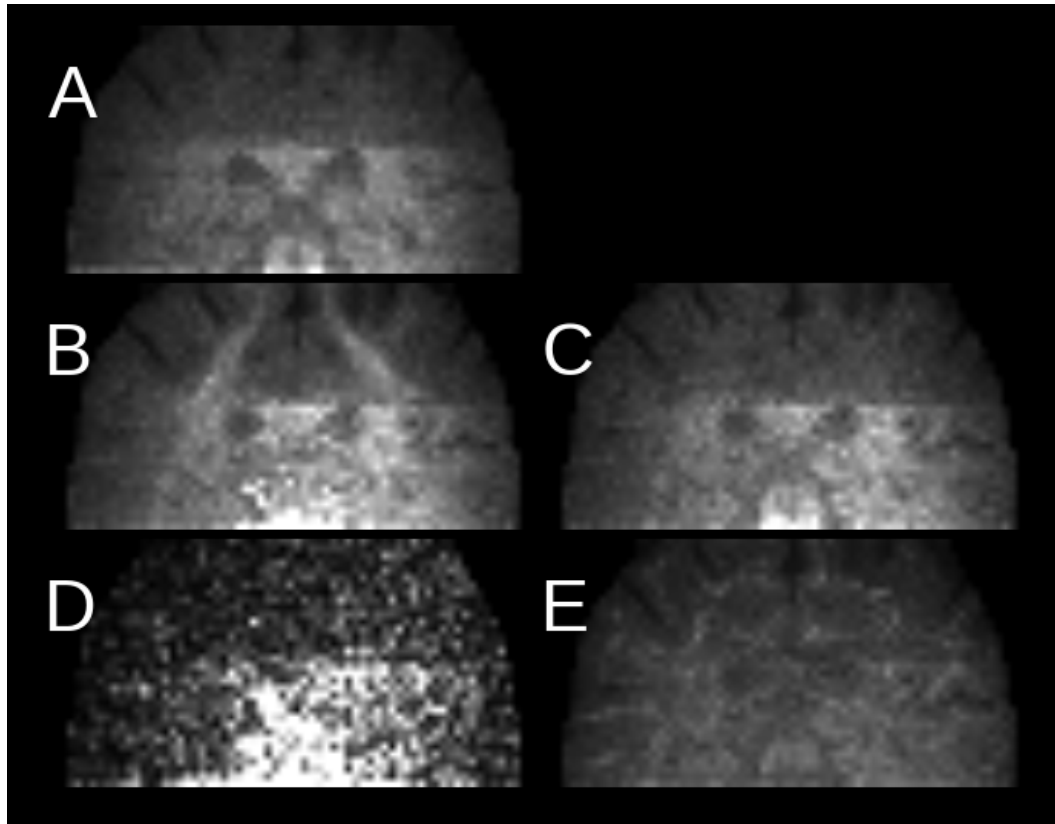


Figure 3.2: The ground truth (A), DTI estimate (B), SH estimate (C), B0 estimate (D) and non-local means estimate(E) noise maps produced from the  $b=3000$  data set.

is a measure that captures both the bias (the difference between the average estimate and the expected result) and the precision (sampling error) in an estimate. From this table we can see that, at least in simulation, the accuracy and precision of the SH residual method is not dependent on  $b$ -value or on SNR. The median noise estimate is close to 1.0 (the simulated noise amplitude) at all three  $b$ -values and at both SNRs. Additionally, the RMS error in the estimate is consistently 0.11 (the analytical value we expect) and does not depend on  $b$ -value or SNR. The DTI model does almost as well as the SH model at a  $b$ -value of 1000; the noise estimates are 1.02 – 1.04 and the RMS error of the voxelwise estimate is 0.11 - 0.12. At higher  $b$ -values, however, the DTI model is insufficient to represent the data and overestimates the noise. At a  $b$ -value of 3000, for example, the median DTI estimates of noise amplitudes are 1.19 and 1.41 at SNR 18 and 25 respectively.

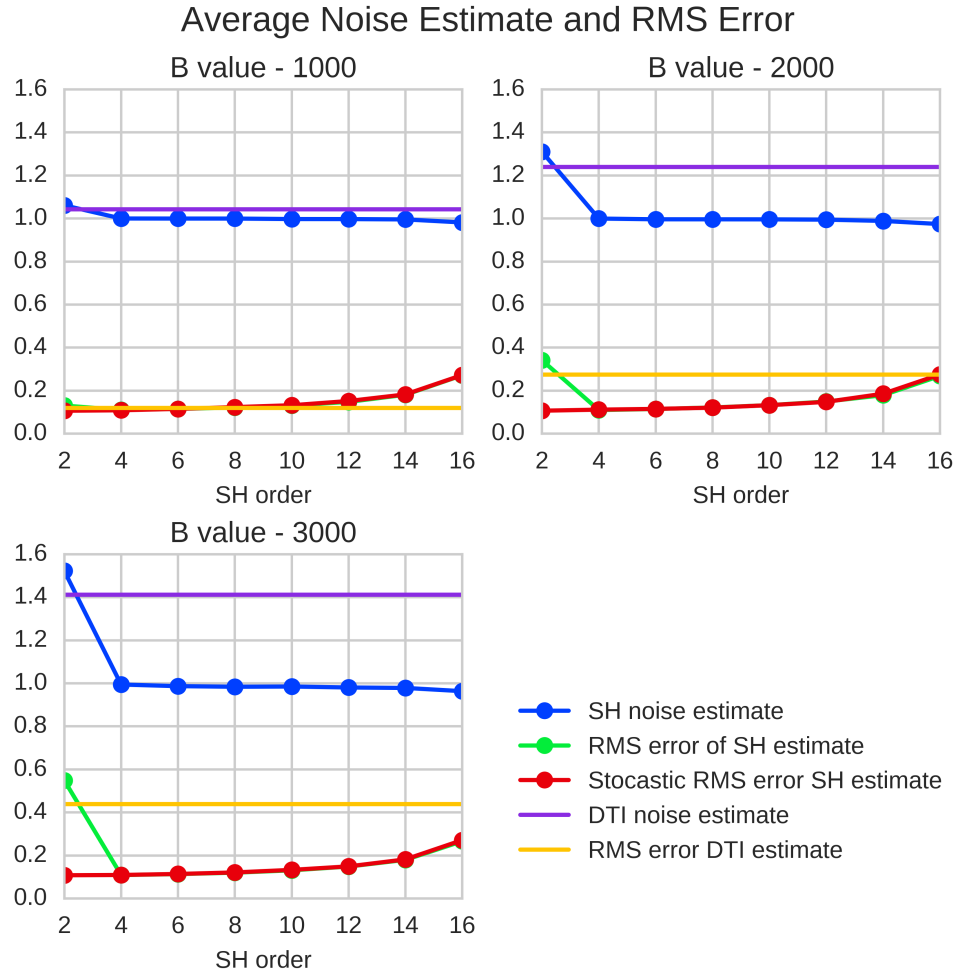


Figure 3.3: Median SH noise estimates and RMS error as a function of SH order computed from simulated diffusion weighted data with SNR of 25. The median DTI noise estimate and RMS error have been included as solid bars for comparison.

### 3.2.2 Human Data

The results using real-world data largely mirror those we see in the simulation. As shown in Table 3.1 and Figure 3.1, the noise estimate using the SH model of order 6 is consistently more accurate and more precise than that of the DTI model; the median estimate is closer to 1.0 and the RMS error is lower. Unlike the simulation, accuracy of the estimate obtained using the SH model does show some b-value dependence, but not as much as the DTI model. The magnitude of this b-value dependence is similar to that observed in the B0 noise estimate. The RMS error in the voxelwise noise estimates obtained using the B0 method is substantially higher than the RMS error associated with the DTI or SH models. While

Table 3.2: The median DTI and SH (order 6) noise estimates and RMS errors computed from simulated diffusion weighted data.

B0 SNR	25	25	25	18	18	18
b-value ( $\frac{s}{mm^2}$ )	1000	2000	3000	1000	2000	3000
DTI noise estimate	1.04	1.24	1.41	1.02	1.11	1.19
DTI RMS error	0.12	0.28	0.44	0.11	0.16	0.23
SH-6 noise estimate	1.00	1.00	0.99	0.99	0.99	0.97
SH-6 RMS error	0.11	0.11	0.11	0.11	0.11	0.11
DWI median SNR	15.28	9.75	6.54	11.08	7.11	4.80

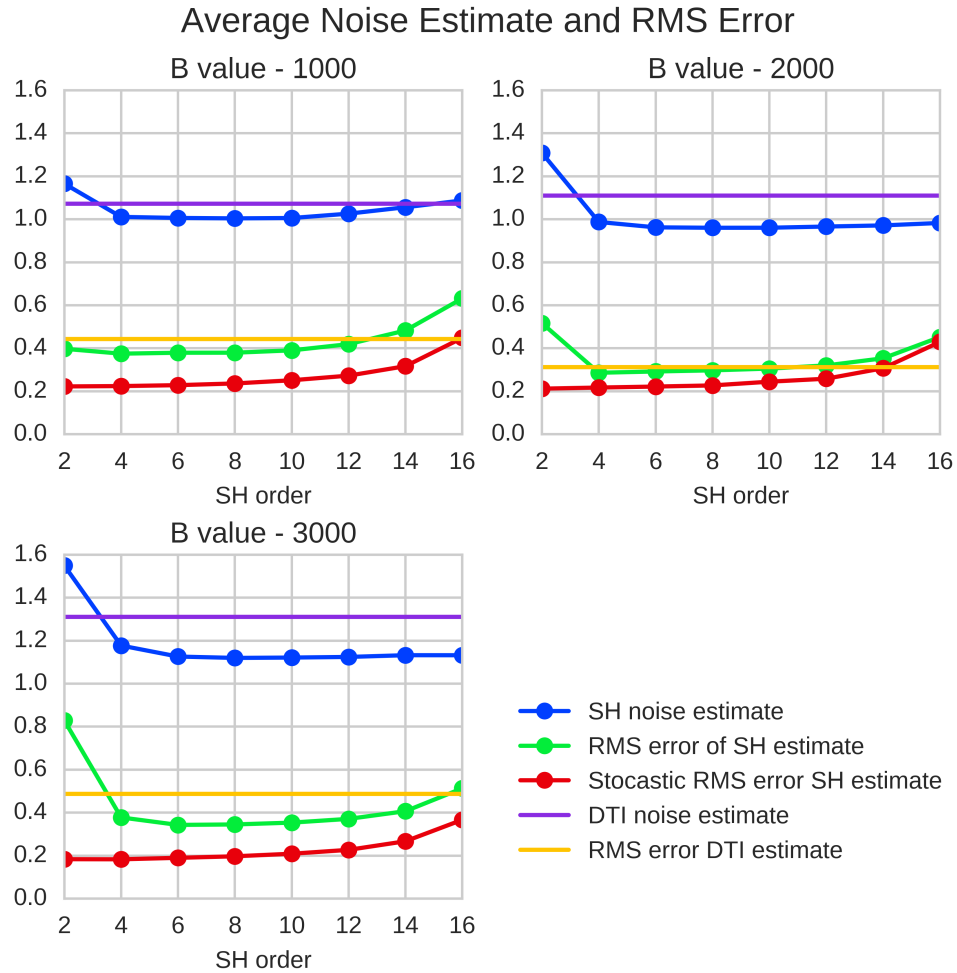


Figure 3.4: Human data median SH noise estimates and RMS error as a function of SH order. The median DTI noise estimate and RMS error have been included as solid bars for comparison. The data was normalized so that the expected noise amplitude is 1.

the B0 method produces a fairly accurate estimate, the average estimated noise amplitude is close to 1.0, the high RMS error means this method does not produce useful noise maps (Figure 3.1 D). The noise maps produced using non-local means filtering overestimated the noise in boundary regions between tissues. The overestimate can be seen in the noise maps and results in a higher RMS error compared to the ground truth noise map. The SH noise estimates and noise maps were the most accurate and most precise, but similar to the simulation results, the SH order influences precision of SH noise estimates.

Figure 3.4 shows the accuracy and precision of the SH noise estimates as a function of SH order. We see that the accuracy (Figure 3.4 blue line) of the SH noise estimate does not change substantially once a sufficiently high SH order is used. In order to measure the precision of the noise, we use the repetition ground truth as a reference. We define the RMS error to be the root mean square error between the noise estimate and the noise amplitude measured from the repetition data set. Lastly, we can estimate the stochastic component of the RMS error using simulations. The RMS error of the noise estimates as a function of SH order (Figure 3.4 green line) predicted the expected stochastic error depicted by the red line. The RMS error of the noise estimates show a minimum at SH order 4 for b-values 1000 and 2000, and a minimum at SH order 6 for b-value of 3000. After the minimum, the RMS error increases with SH order. The RMS error in the SH estimates (Figure 3.4 green line) was higher than the stochastic RMS error we predicted using simulation (Figure 3.4 red line). This means that not all the error in the noise estimate can be explained by sampling variation, and other sources of error are present in the estimate. However, the two lines show the same trend and sampling variation does explain the increase in RMS error and decrease in estimate precision as a function of SH order.

The precision difference between the noise estimates produced using low and high order SH models can be seen by comparing the respective noise maps. Figure 3.5 shows axial and coronal slices from three noise maps estimated using SH orders 4 (B), 14 (C) and 16 (D) compared to the ground truth noise map generated from repeated measurements (A). The SH order 6 (B) estimate is visibly smoother than higher order estimates and more fine detail is visible in this lower order noise map. For example, the outline of the Globus Pallidus (in the red box) is visible in the ground truth noise map (A) and this structure has visibly lower noise amplitude than the surrounding tissues. This structure has similar detail in the SH order 6 estimate (B), but in the SH order 14 (C) and 16 (D) estimates the structure becomes progressively harder to resolve. This is not surprising because the precision of the noise estimates defines the level of contrast in the noise maps. The SH order 6, 14 and 16 estimates have 152, 60, and 27 degrees of freedom respectively. SH harmonic order 8 is commonly used to fit SH functions to HARDI data. In order to produce noise maps with

a matching number of degrees of freedom to B, C and D in Figure 3.5 with a SH order 8 model, one would need to acquire 198, 106, and 73 linearly independent diffusion gradient directions respectively.

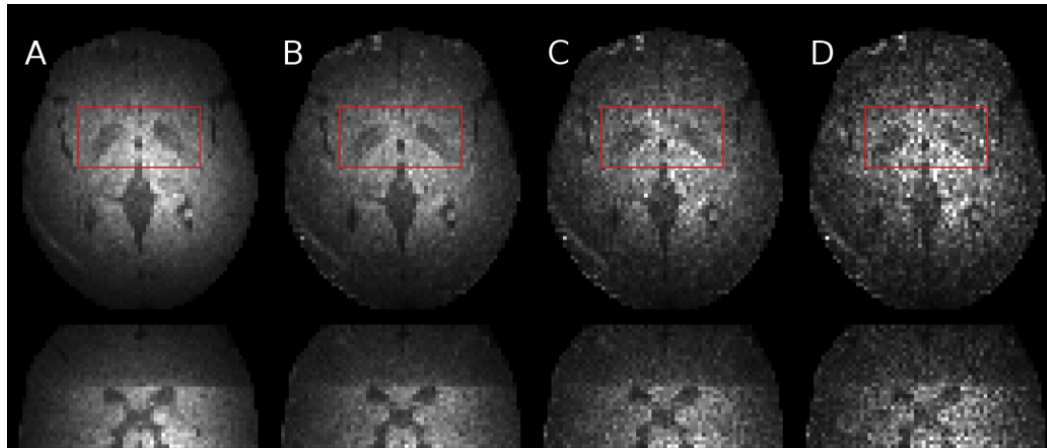


Figure 3.5: Noise maps produced from DWI data. From left to right A) noise amplitude measured using repeated samples B) SH order 4 noise map estimate C) SH order 14 noise map estimate D) SH order 16 noise map estimate.

### 3.3 Discussion

Our results show that using a SH model to estimate noise maps for HARDI data is more accurate and more precise than using a DTI model or using the currently available  $b_0$  and non-local means methods. The accuracy of the noise map is mostly independent of the SH order used, provided that a sufficiently large order is used. However, the precision of the estimate does depend on the choice of SH order. Even though the SH model noise estimate is better than the tensor or  $B_0$  estimates, we did observe some discrepancy between SH noise estimate and the ground truth noise measured by repeated data acquisition, not all of which can be explained by stochastic variation. The simulation we used to estimate the expected stochastic RMS error is conservative and we would not expect to achieve that level of precision, even under ideal conditions. However, we believe the precision of the noise maps may be improved if better eddy current correction and image alignment methods are used in the processing of the raw data. The repeated gradient acquisition data we used for our ground truth noise measurement has very consistent alignment. The consistent alignment is not surprising because each of the repeated images is impacted by the same eddy current distortions. On the other hand, the data used to produce the bootstrap noise estimates has

visibly worse alignment between the images. The areas where misregistration is most pronounced are consistent with the areas of largest differences between the noise estimate and the ground truth. While we explored possible parameters for improving the alignment, the simple eddy current correction method we used is not capable of fully correcting the eddy current distortions present in this kind of data.

Other factors may also contribute to differences between the ground truth noise measurement and the bootstrap noise estimates, but it is important to note that there is no evidence that model failure is one of these factors when sufficiently high SH order is used. This is in contrast to the DTI method, where there is strong evidence for model failure. In the presence of model failure, the bootstrap procedure overestimates the residuals and overestimates the variance in the metric of interest. In this case, we would expect a model failure to result in a positive bias, or an overestimate, of the noise amplitude. The DTI results show an overestimation of the median noise. The noise maps show that this overestimation is not uniform over the image, but most predominant in the white matter regions where we expect crossing voxels. Additionally, the SH model produces a lower noise estimate in these regions, which is consistent with the expectation that the SH model can better represent crossing voxels. The SH model shows evidence of model failure at order 2 and order 4 for  $b=3000$ . However, there is no evidence of model failure at higher SH orders. In the presence of model failure, we expect the median noise estimate and RMS error to decrease as higher SH order models decrease the model failure.

In this study, we have focused on noise estimation for single shell q-ball data, however these methods could still be useful for studies that acquired multi-shell diffusion acquisitions. The noise in each shell could be analyzed independently using these methods as long as there were at least about 20 gradients acquired in each shell. This approach is the simplest and would work for acquisitions where imaging parameters are adjusted independently for each shell to maximize SNR at each shell. The disadvantage of treating each shell independently is that model parameters must be fit resulting in a noise estimate with fewer degrees of freedom. To avoid this one could use one of the several recently published methods that use SH functions to fit multi-shell data [24]. Because these methods make additional assumptions about the nature of the data being modeled, they would need to be validated independently, but that work is outside the scope of this paper.

Understanding which combination of model and parameters produces the best voxel-wise noise map is important not only for accurately estimating noise, but also for choosing the right bootstrap procedure in other applications. In recent years, many new models for characterizing diffusion profiles in brain tissue have been proposed. Many of these models, such as NODDI[38] or CHARMED[39], require complex non-linear fitting of the diffusion

data. Residual bootstrap methods could be used to estimate the standard errors produced by these models, but bootstrapping the models directly is not practical because leverage correcting the residuals produced by these models is impractical or impossible. However, one could use a different model to produce bootstrap datasets and then apply the model of interest to those datasets in order to estimate standard errors in the model parameters or metrics. The quality of an estimate produced using a bootstrap procedure depends on how well that procedure can generate bootstrap samples that are representative of the distribution from which the data is sampled. Any bootstrap sample that represents the underlying data distribution should, at minimum, be able to estimate the variance (or noise) of that distribution. Therefore, when picking a bootstrap procedure for any dMRI application, one should consider how well that procedure estimates the imaging noise.

While this study demonstrates that noise maps can be estimated from dMRI acquisitions, how well the methods presented here perform may depend somewhat on the MRI acquisition sequence used and the biological characteristics of the tissue being imaged. Two important criteria are required for model residual bootstrap methods to be effective. First, the model must be a reasonably accurate representation of the diffusion characteristics of the biological tissue. Our results indicate that the tensor model fails at high b-values, causing the noise to be overestimated. While the SH model has similar accuracy over a b-value dependent range of orders, the lowest order in that range provides best precision in the noise estimates. The second criterion for accurate noise estimation is that the imaging noise must be approximately normally distributed. MRI imaging noise is generally believed to have a Rician distribution that can be approximated as normal when the SNR is sufficiently high. Therefore this criterion is satisfied for most dMRI data sets. However, data sets that approach the noise floor will result in some bias in the noise estimates due to violation of the normally distributed noise assumption.

### **3.4 Conclusion**

We have demonstrated that model residual bootstrap methods can be used to produce noise maps for HARDI data sets and are superior to the currently used methods using b0 repetitions and nonlocal-means. For this purpose, the DTI model only performs well if relatively low b-values are used. The SH model performs well at all b-values, but the precision of the SH noise estimate depends on the SH order used. In our simulations and human diffusion data, SH orders 4 and 6 provided the best balance between fitting the data well and preserving degrees of freedom for the noise estimate.

## CHAPTER 4

# Dipy, a Diffusion Analysis Platform

Fiber tracking, or tractography, is a developing set of techniques for identifying and measuring the properties of white matter (WM) pathways in mammalian brains. By measuring the restricted Brownian motion of water molecules in brain tissues, diffusion MRI gives us a way to examine the microstructural properties of those tissues. Diffusion models such as diffusion tensor imaging (DTI) or constrained spherical deconvolution (CSD) can be applied to diffusion MRI data sets to estimate tissue properties from diffusion weighted images. These tissue properties allow us to see into the microstructure of the axons that make up white matter tissues. These axons, and their surrounding myelin, restrict water diffusion therefore diffusion weighted MRI is sensitive to their diameter, density, integrity, and directionality. Fiber tracking allows us to take this analysis to the next level and begin to estimate the properties of individual WM tracts or the collections of tracts that make up the structural network of the brain. Dipy provides a unified tool set which allows researchers and developers to both analyze diffusion MRI data sets and develop new modeling and analysis methods. This chapter describes the design and organization of dipy's diffusion modeling and fiber tracking tools and how these tools can be extended to implement analysis methods.

### 4.1 Model the diffusion signal

The first step in any diffusion MRI analysis, after pre-processing is complete, is to model the diffusion MRI signal. The model that one chooses depends on several factors. As with most modeling applications, the right model will balance several competing priorities including: simplicity of the model, assumptions built into the model, and accuracy of the result. In diffusion MRI, finding the optimal balance largely depends on the characteristics of the data to be modeled and the specifics of the application. For example, when the diffusion data to be modeled includes images acquired with different diffusion weightings, often



called multiple q-shell data, a more complex model like Multitissue Constrained Deconvolution or Diffusion Spectrum Imaging can be used. If the data is acquired using a large number of diffusion gradient directions, more than 50, on a single q-shell with moderate to high b-value, then an intermediate model like QBall or Constrained Spherical Deconvolution might be the best choice. For data sets with a low b-value or a relatively few gradient directions, than a more simple model, such as DTI, is most appropriate.

It's also important to consider the requirements of the application when picking a model. While a given data set might support more complex modeling, if the goal of the project is identify large, well defined white matter tracts in the brain, then the additional complexity and computational cost associated with a more complex model might not be justified. However, if the goal is to model areas of the cerebral white matter with finer structures and axonal crossings, than a simple model like DTI will simply fail to capture the underlying complexity of the biological architecture. The choice of model is often a subtle decision that can only be made empirically. Dipy aims to aid the user in the task of choosing the right models by providing a model API so that users can run the same tracking and analysis methods using different models in order to compare results.

### 4.1.1 Model and Fit Classes

Models in dipy serve two related functions. First a model can be used to fit diffusion MRI signals and estimate model parameters. Second, some dipy models provide a `predict` method, which allows the model to predict diffusion MRI signals from model parameters. Here is a formal description of the model and fit interfaces in Dipy.

1. The model class.
  - (a) Models classes are initialized with any required parameters to create model instances. In this context, a gradient table is considered a parameter and should be passed to the model in the constructor.
  - (b) Models should have a `fit` method. The `fit` method should return a fit object, meaning an instance of a fit class.
    - i. The `fit` method should take two parameters: `data` and `mask`. `Data` is an array of diffusion MRI signals and `mask` is a binary array where True values identify the non-background voxels of the image.
  - (c) The model may have a `predict` method. The `predict` method should predict diffusion signals from model parameters.

- i. The `predict` method should take two arguments: model parameters and a gradient table. The gradient table argument should be optional and the gradient table associated with the model should be used, by default.

1. The fit class.

- (a) Each fit object should have a `model` attribute which will generally be the model object used to create the fit object.
- (b) The fit class may also have a `predict` method. The `predict` method of the fit class does not take model parameters as an argument, instead it estimates a signal prediction based on the model parameters associated with the fit instance.
- (c) The fit object may have an `odf` method which takes a `Sphere` instance as an argument and returns an the Orientation Distrubtion Function estimated by the model and sampled on the vertices of the `Sphere` instance. The `odf` method is used for peak finding and fiber tracking.
- (d) The fit class may have attributes and methods specific to the model type which the fit represents. For example, a DTI fit has the attributes `fa` and `md` which are the fractional anisotropy and mean diffusivity, respectively, of the diffusion tensor.

All the models in `dipy` follow this basic interface. `Dipy` only enforces this interface through "duck typing", meaning the interface is checked at run time on an as needed basis, as opposed to strictly enforced using a mechanism such as inheritance. The result of this design choice is that `Dipy` does not require all model classes to implement the entire model interface. However, a model class that omits part of the interface may fail if it interacts with `dipy` functions or methods which require the missing portion of the interface.

## 4.2 Dipy Utilities for Models

### 4.2.1 Gradient Table

`Dipy` provides some tools for making the implementation of new models easier. The first of these tools is the `GradientTable` class. A gradient table object can be initialized from gradient table files, for example `bvec` and `bval` files commonly used in diffusion imaging, or simply an array of gradients by using the `gradient_table` factory function in `dipy.core.gradients`. A gradient table object allows a user to easily access different representations of gradient information. For example, the `bvals` attribute of the

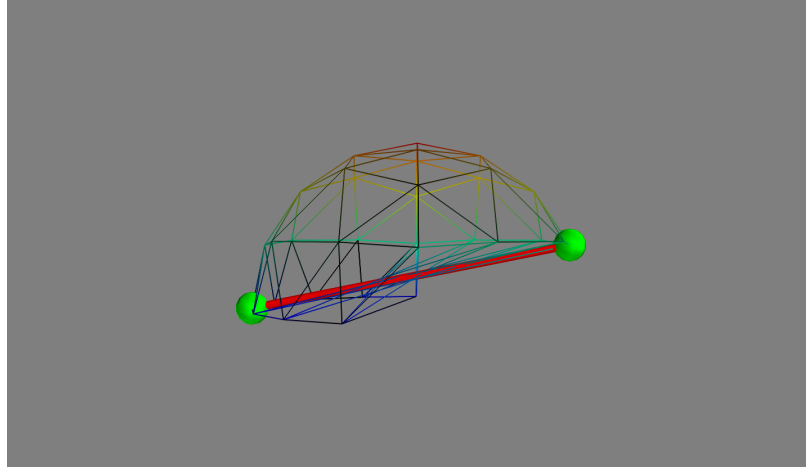


Figure 4.1: A HemiSphere with 66 vertices. The faces of this sphere are shown as a wire mesh. Two of these vertices have been highlighted in green. These two vertices are neighbors, even though they appear on opposite sides of the spherical surface, and therefore are connected by an edge. The edge that connects these two vertices is highlighted in red.

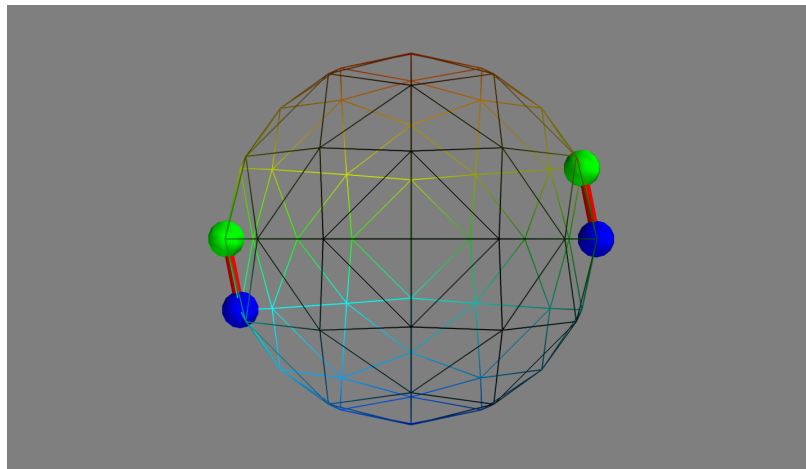


Figure 4.2: A Sphere which is the mirror of the HemiSphere in figure 4.1. This Sphere has a total of 132 vertices. The green vertices of the original HemiSphere have been mirrored to create the blue vertices. Each of the green points is now a neighbor too, and shares an edge connection with, the mirror of the other green point.

gradient table object exposes the diffusion weighting, commonly known as the b-value, of each gradient. Similarly, the `bvecs` attribute exposes the direction of each gradient as a unit vector and the `gradients` attribute exposes the total gradient vector.

## 4.2.2 Spheres and Hemispheres

Dipy uses the `Sphere` and `HemiSphere` classes as discrete representations of the unit sphere. The `HemiSphere` class is a subclass of the `Sphere` class, provides the same interface, and can be used anywhere Dipy requires a `Sphere` instance. Both `Sphere` and `HemiSphere` objects consist primarily of a set of dispersed points, or vertices, on the unit sphere. The `Sphere` class also provides `faces` and `edges` attributes which provide neighbor information about the vertices of the sphere. Each face of the `Sphere` is triplet of integers between 0 and  $N - 1$ , where  $N$  is the number of points of the `Sphere`. The faces, taken together, make up a mesh surface which approximates a sphere. Each edge of a `Sphere` is a pair of integers between 0 and  $N - 1$  and represents two neighbor points. Each point has a neighbor relationship with the closest points to itself. This neighbor relationship is important for peak finding and fiber tracking applications.

Though the `Sphere` and `HemiSphere` classes are similar, there is one key difference. The `HemiSphere` class incorporates an understanding of antipodal symmetry into its definition of distance between points. On the surface of the sphere, we can define a distance function between two points,  $\vec{v}_1$  and  $\vec{v}_2$  as  $d(\vec{v}_1, \vec{v}_2) = \cos^{-1}(\vec{v}_1 \cdot \vec{v}_2)$ . This is the intuitive definition of distance that is familiar to most people. Under this definition, the points  $\vec{v}$  and  $-\vec{v}$  are separated by a distance of  $\pi$ , the furthest possible distance between two points on a unit sphere. In the context of diffusion imaging, we want to be able to represent antipodally symmetric distance function where the points  $v$  and  $-v$  are in effect the same point. To represent this symmetry, we can define a new distance function on the sphere,  $d(\vec{v}_1, \vec{v}_2) = \cos^{-1}(|\vec{v}_1 \cdot \vec{v}_2|)$ . By using this new distance function, the `HemiSphere` class is able to effectively represent an antipodally symmetric sphere with half as many points, without losing important neighbor relations. While it is redundant to use a full `Sphere` to represent an antipodally symmetric spherical surface, sometimes a full `Sphere` object can still be useful, for example for visualizing spherical functions without discontinuities. The `HemiSphere` class provides a `mirror` method to duplicate and project the vertices of a `HemiSphere` onto a full sphere. Figures 4.1 and 4.2 show a 3d rendering of a `HemiSphere` and its mirror respectively.

Dipy provides a `default_sphere` object, which is an instance of `HemiSphere`, in `dipy.data`. The `default_sphere` consists of 362 vertices dispersed on a unit hemisphere. We've found that this number of points provides a good balance between coverage and computational performance.



Figure 4.3: The organization of the local tracking components. Green arrows denote inputs into tracking components.

## 4.3 Local Tracking Components

By framing the local tracing problem as one of independent but interacting modules, we're able to focus on the behaviour of the independent components. In this section we discuss how the components of the local tracking model are built, the responsibilities of each component and how the components interact.

### 4.3.1 Direction Getter

As the name suggests, it is the responsibility of the direction getter to pick tracking directions based on the tissue information at the current tracking location. The first current tracking location is the last point, so far, of the streamline being tracked given relative to the diffusion MRI images. The second piece of available information is the direction of previous streamline segment. This second piece of information is important because it allows crossings, and other complex architectures, to be modeled by the tracking algorithm. When multiple plausible tracking directions are available, the best directions can be chosen contingent on the direction of the previous segment. The direction getter must also be able to provide initial directions for tracking from seed points. Because seed points are used to initialize the tracking procedure, there is no previous segment from which to extract a direction. A direction getter can provide multiple initial directions for a given seed point, for example if the seed point is in a WM structure where axons from different tracts cross. The

tracking framework is then responsible for either picking the best of the initial directions or tracking multiple directions and returning a streamline for each of the directions.

Dipy provides several standard direction getters. The function `peaks_from_model`, from `dipy.direction` returns a deterministic direction getter which estimates tracking directions from ODF or FOD peaks. Both q-ball type, which typically estimate ODFs, and spherical deconvolution type, which typically estimate FODs, can be used with this direction getter. Dipy also provides a probabilistic direction getter, `ProbabilisticDirectionGetter`

in `dipy.probabilistic_direction_getter`. This direction getter returns directions stochastically by choosing from a discrete set of predefined directions based on a set of weights. The `ProbabilisticDirectionGetter` is usually constructed using the FODs as those weights. While in principle any function defined on the surface of the sphere can be used to weight the selection probabilities of the tracking directions, ODF functions tend to be too smooth and produce poor tracking results. For that reason, the FODs of spherical deconvolution type models, or similar functions, are typically used with the `ProbabilisticDirectionGetter`.

Dipy also allows users to define their own direction getter class, either in python or in cython. Here are the requirements for implementing a direction getter.

1. A direction getter should inherit from `DirectionGetter` from the module `dipy.tracking.local`.
2. A direction getter should implement a `get_direction` method.
  - (a) The `get_direction` method takes two arguments, `point` and `direction`.
    - i. Both arguments are of type `memoryview` with shape `(3,)` and format `'d'`.
    - ii. `point` is the current streamline position in voxel coordinates.
    - iii. `direction` is the direction of the previous tracking step, given as a unit vector.
  - (b) `get_direction` should return 1 if no tracking direction can be established and 0 otherwise.
  - (c) `get_direction` should update `direction` with the next tracking direction if 0 is returned.
3. A direction getter should implement an `initial_direction` method.
  - (a) The `initial_direction` method takes one argument, `point` similar to `get_direction`.

- (b) The `initial_direction` method should return an  $(N, 3)$  array of directions as unit vectors.
  - i. These directions should be suitable tracking directions for tracking from seeds at `point`.
  - ii. Because of the antipodal symmetry in diffusion imaging, the directions  $x$  and  $-x$  are considered the same. Only one of the directions associated with each pair should be included.
  - iii. If some directions are more suitable for tracking, the directions should be sorted in order of suitability. The most preferred direction first.
  - iv. If no suitable directions can be established, a  $(0, 3)$  array should be returned.

### 4.3.2 Tissue Classifier

The tissue classifier is the component of the local tracking model which is aware of the different tissue types in the image. At its most basic, it is the responsibility of the tissue classifier to terminate streamlines when they leave WM tissues. However, the tissue classifier can also be used to exclude streamlines which pass through tissues which are known not to contain axons. Currently `dipy` supports four point classes, `TRACKPOINT`, `ENDPOINT`, `INVALIDPOINT`, and `OUTSIDEIMAGE`. The choice of tissue classifier when tracking depends on the imaging and other data available for assessing tissue types. Based on all the available information about anatomical tissues, the tissue classifier must be able to determine which of the above point classes best represents each tracking point.

`Dipy` currently has several standard tissue classifiers available. The threshold classifier, `ThresholdTissueClassifier` from the module `dipy.tracking.local`, allows metric maps to be used, along with a threshold value, to classify tissues as either WM or not WM. This classifier is a good choice when the user does not have or want to use imaging data in addition to the diffusion MRI data set. In this case, simple metrics like FA or generalized fractional anisotropy (GFA) can identify WM tissues reasonably well. The binary classifier, `BinaryTissueClassifier` from the module `dipy.tracking.local`, allows WM segmentations to be imported as binary masks. This option is a good choice when additional imaging data or better analysis tools are available for producing a white matter segmentation. For example, `freesurfer` [40] or FSL's `FAST` [41] can be used to identify WM tissues from structural images, and those results can be used for fiber tracking. Lastly, the Anatomically-Constrained Tractography (ACT) classifier, `ActTissueClassifier` from the module `dipy.tracking.local`, al-

allows the use of two probabilistic tissue maps, an include map and an exclude map. This approach allows for a more fine grained classification of streamline points and allows some streamlines to be excluded completely based on the tissues traversed. This classifier is based on work by Smith et. al. [42] and Girard et. al. [43].

Dipy also allows a user to define their own tissue classifiers. Here are the requirements for implementing a direction getter

1. The tissue classifier must inherit from `TissueClassifier` from the module `dipy.tracking.local.tissue_classifier`
2. The tissue classifier must implement a `check_point` method.
  - (a) The `check_point` method takes one argument, `point`.
    - i. `point` is of type `memoryview` with shape `(3,)` and format `'d'`.
    - ii. `point` is the current streamline position in voxel coordinates.
  - (b) The `check_point` method should return a `TissueClass`.
    - i. Currently the four valid tissue classes are: `TRACKPOINT`, `ENDPOINT`, `INVALIDPOINT`, and `OUTSIDEIMAGE`.

### 4.3.3 Local Tracking Class

The local tracking class implements the basic logic of the local fiber tracking algorithm. In the heart of the local tracking class is the main tracking loop. The main tracking loop uses a direction getter, a tissue classifier, and a seed to produce half a streamline. Because of antipodal symmetry, each seed is tracked in two opposing directions and the results joined to produce each streamline. Appendix ?? contains psudo-code for the local tracking algorithm including the basic loop. In dipy, the basic tracking loop is implemented using cython. Because this is the very inner loop of the tracking algorithm, it is important that this part of the code be optimized for performance. By implementing this loop in cython, we're able to compile this piece of the code and call the compiled code from python. Using cython has another major advantage. Because the abstract base classes `DirectionGetter` and `TissueClassifier` are both also defined in cython, their subclasses can implement the direction getter and tissue classifier interfaces either in cython or in python. While implementing these subclasses in python will be slower than using an optimized cython implementation, this design gives users maximum flexibility. For development and testing of new concepts, developing in python allows users to quickly implement and try out different ideas and for many basic use cases, python is often fast enough. However, for performance critical tasks users can implement direction getters and tissue classifiers in cython.



### 4.3.4 Multi Voxel Fit Decorator and Class

When fitting a dipy diffusion model, a single voxel or an entire data set can be passed to the model's fit method. While this design choice makes it easier for users to create and fit models to their data, it requires that all model and fit classes are able to accept and work with 4d data sets. Some models might choose to explicitly handle the 4d data sets, however in some cases it's helpful to be able to abstract out the looping logic and implement the model with respect to only one voxel. Dipy provides a decorator, a python object used to modify function or method definitions, which allows users and developers to easily implement a model without explicitly considering the 4d nature of the data. This decorator, `dipy.reconst.multi_voxel.multi_voxel_fit`, can be applied to the fit method of a model class. The decorator guarantees that the fit method of the model is called for each data voxel when a multi-voxel data set is passed to the fit method. The implementation of the fit method should return a fit class, like all other models. Here is an example of a model and fit class pair that use the multi-voxel decorator. Notice that other than the decorator, both classes can ignore the multidimensional nature of the data.

```
from dipy.reconst.multi_voxel import multi_voxel_fit, MultiVoxelFit

class SampleModel(object):
    def __init__(self, gradient_table):
        self.gtab = gradient_table

    @multi_voxel_fit
    def fit(self, data):
        return SampleFit(self, [data.mean(), data.var()])

class SampleFit(object):
    def __init__(self, model, fit_parameters):
        self.model = model
        self.mean = fit_parameters[0]
        self.var = fit_parameters[1]
```

A fit method with is decorated with the multi-voxel fit decorator can return either an instance of the appropriate fit class if in the data input is a one dimensional array of diffusion signals or an instance of the `MultiVoxelFit` class if the input is a multidimensional data set. The `MultiVoxelFit` class is an abstraction which allows the attributes and methods of the single voxel fit class to be applied to man fit instances at once. For example, the sample fit class in the previous example has `mean` and `var` attributes that are scalars in the single voxel case. In the multidimensional case, the fit method returns a `MultiVoxelFit`.

The `mean` and `var` attributes now return multidimensional arrays which represent the mean and variance computation for each voxel of the multidimensional data set that was passed to the `fit` method.

```
import numpy as np

gradient_table = object()
model = SampleModel(gradient_table)
sample_data = np.random.random((10, 10, 10, 64))

fit = model.fit(sample_data[0, 0, 0, :])
assert np.isscalar(fit.mean) and np.isscalar(fit.var)

fit = model.fit(sample_data)
assert type(fit) is MultiVoxelFit
assert fit.mean.shape == (10, 10, 10)
assert fit.var.shape == (10, 10, 10)
```

This decorator is used in the implementation of several `dipy` models and allows users to quickly and easily implement their own diffusion models.

## 4.4 Streamline Utilities

Tractography can be used to identify connections between brain regions, estimate the path of specific connection or measure white matter tissue properties in a single subject brain. To facilitate these applications `Dipy` provides several utilities that transform streamline data. This section describes some of these utilities in more detail.

### 4.4.1 Connectivity Matrix

A connectivity matrix represents the streamlines that begin and end in two given tissue regions of the brain. To create a connectivity matrix one needs to start with a parcellation of the brain tissues. In this parcellation each tissue region is represented by a unique, non-negative integer value. The zero value is usually reserved for background and non-brain tissues. This tissue parcellation can then be combined with the result of the local tracking output to produce a connectivity matrix. This example shows a typical use of the `connectivity_matrix` function. Here we return both the matrix, the number of streamlines that connect two regions, and the mapping. The mapping is a dictionary of the streamlines that connect every pair of regions. Each streamline is repre-

sented in the matrix and the mapping exactly once, and for every tissue region  $i$  and  $j$ , `matrix[i, j] == len(mapping[i, j])`.

```
from dipy.tracking.utils import connectivity_matrix

matrix, mapping = connectivity_matrix(streamlines, parcellation, affine=
    affine,
                                   symmetric=True, return_mapping=
                                   True,
                                   mapping_as_streamlines=True)
```

## 4.4.2 Targeting

Targeting is the filtering of streamlines based on whether or not they intersect with a specific ROI. Multiple ROI targets can be combined by sequential application of the target function. This example demonstrates how one could select only streamlines that intersect with both of two given ROIs: ROI1 and ROI2.

```
from dipy.tracking.utils import target

# select streamlines that intersect with both ROI1 and ROI2
intermediate_result = target(streamlines, ROI1, affine=affine)
filtered_streamlines = target(intermediate_result, ROI2, affine=affine)
```

To exclude streamlines based on a ROI, the `include` keyword should be set to `False`. This example shows how one can select streamlines that intersect with ROI1 but do not intersect with ROI2.

```
from dipy.tracking.utils import target

# select streamlines that intersect with both ROI1 but do not intersect
# with ROI2
intermediate_result = target(streamlines, ROI1, affine=affine)
filtered_streamlines = target(intermediate_result, ROI2, affine=affine,
                             include=False)
```

More complex targeting and filtering procedures can be implemented by applying logical operators to the ROIs, as well as chaining together multiple targets.

## 4.4.3 Moving Streamlines

Streamlines represent paths through three dimensional space. Being able to describe these paths in different coordinate spaces is often useful. For example, if a subject returns for

a followup study, one may want to co-register the streamlines from the first session with the MRI data of the second session. In order to move streamlines between two coordinate spaces, one needs to know the affine transformation between each of the coordinate spaces and a common reference space. Sometimes, one of the two coordinate spaces is the same as the reference space, in that case one of the two affine transformations is simply the identity transform. Dipy provides a utility function that will transform streamlines between two coordinate spaces given a common reference space.

```
from dipy.tracking.utils import move_streamlines

output_streamlines = move_streamlines(streamlines, output_space=
    output_affine, input_space=input_affine)
```

#### 4.4.4 Density Maps

A tractogram can be used to create a density map, sometimes called a track density image or TDI[44, 45]. Because the streamlines of a tractogram are defined in a 3d continuous space, density maps can be created at arbitrary resolution. When the resolution of the track density image is higher than the resolution of the diffusion MRI data used to create it, the density image is called a super resolution TDI [45]. If the voxel size of TDI is smaller than the step size of the streamlines in the tractogram, subsegmenting the streamlines will produce a better quality TDI. In this example we combine the `subsegment` and `density_map` functions to create a super resolution TDI.

```
from dipy.tracking.utils import subsegment, density_map

data_matrix = [128, 128, 64]
tdi_matrix = [1024, 1024, 512]
tdi_resolution = .125 # approximate resolution in mm

super_res_streamlines = subsegment(streamlines, tdi_resolution)
super_res_tdi = density_map(super_res_streamlines, tdi_matrix,
    affine=super_res_affine)
```

## CHAPTER 5

### BMI as it Relates to White Matter Tissues

The negative health effects of being overweight or obese are well known and include higher risk of cardiovascular disease, diabetes and even some cancers. The neurological and cognitive effects of having high body weight are less well known. Obesity has been associated with both lower levels of cognitive function and an increased risk of dementia later in life. Recent work has found that obesity and high body mass index (BMI) are associated with declines in several cognitive domains including executive functioning, attention and motor skills [46, 47, 48, 49]. Additionally, persons with high BMI in mid-life have a higher risk of developing both Alzheimer's Disease and non-Alzheimer's dementia. Even children with higher BMI appear to have declined cognitive abilities compared to their normal weight peers [50], and among college athletes with excellent cardiovascular health, high BMI is associated with lower cognitive function [51]. However, there is also evidence that higher body weight may play a neuro-protective role later in life [52]. Brain imaging can help us better understand the mechanisms that underlie these links between body weight and cognition.

In addition to cognitive effects of body weight, recent work has demonstrated that high body weight can effect the macro and micro-structure of the brain. Lower brain volumes have been reported in people with high BMI [53, 54]. Additional work focused on regional differences in brain volumes has found both gray matter (GM) and white matter (WM) atrophy, particularly in the temporal and parietal regions of the brain [55]. Micro-structural alterations in brain tissues associated with high BMI and obesity have also been reported. Several studies using diffusion tensor imaging (DTI) have found lower fractional anisotropy (FA) and possibly lower axonal coherence in several white matter tracts, including the corpus callosum and cingulate bundle [56]. These DTI findings are supported by a spectroscopic imaging study which found that high BMI correlated with lower concentrations of N-acetylaspartate, a marker of neuronal viability [57]. While these findings demonstrate that the observed neuro-cognitive differences between persons with high BMI

and normal weight are mediated through alterations in the brain, there is disagreement in the literature about which brain regions are implicated and to what extent.

In this work we apply a new methodology for examining the associations between white matter biomarkers and patient outcomes to the a large, public data set of young subject in order to better understand the link between body mass and brain health. This new methodology maps white matter tissue markers onto a large, intersecting set of white matter pathways reconstructed using advanced tractography methods. The tract based markers are then statistically modeled to identify associations between white matter structures and health outcomes. We present the results of our white matter analysis along with cortical thickness analysis and show that the BMI effects to the brain, in this young population, are most measurable in the white matter tissues.

## 5.1 Methods

For this study we used the S900 release of HCP [58]. This release included 766 subjects with complete diffusion MRI data sets and BMI information. The mean age of the group was 28.8 +- 3.7 years old and the group had 426 females and 339 males. The sex of 1 subject was not disclosed. The group had a mean BMI of 26.4 +- 5.1. The HCP acquires 3 similar diffusion MRI scans with 1.25mm isotropic resolution. Each of the 3 scans have different gradient directions so they can be combined into one data set with 270 unique gradient directions. These 270 directions are distributed over 3 shells, with b-values of 1000, 2000 and 3000. The combined data set also contains 18 volumes with minimal diffusion weighting (b0 images). Both unprocessed and preprocessed versions of the diffusion MRI images are available publicly. For this study we used the preprocessed images. These images were reconstructed using a SENSE1 MRI image reconstruction and then eddy corrected using a gaussian process predictor [59]. In addition to diffusion MRI images, we also used the structural T1 images, the MNI registrations, and the freesurfer segmentations of those T1 images provided as part of the structural preprocessed HCP data set [60]. We chose the HCP data for this study because it contains a large number of patients, high resolution structural imaging, freesurfer processing, and high resolution and high quality diffusion MRI images. This data is also publicly available for repetition and follow up studies.

### 5.1.1 Fiber Tracking and Diffusion Metrics

From the diffusion MRI data we extracted DTI metrics for a large number of white matter pathways identified using whole brain fiber tracking. These pathway specific metrics served as features in our statistical analysis. In this study we focused on two metrics specifically, fractional anisotropy (FA) and mean diffusivity (MD). Our goal was to find the mean FA and mean MD along as many white matter pathways as possible. To achieve this goal we used whole brain tractography to create a tractogram for each subject using fiber tracking tools in the Dipy software library [31]. Our fiber tracking methodology is described below. We used probabilistic fiber tracking with a step size of .5 mm to generate tractograms for each subject. We created gray matter and white matter masks using the freesurfer segmentation results provided in the HCP data set. The probabilistic fiber tracking was seeded in the white matter voxels adjacent to gray matter tissues using 27 seeds per voxel. The probabilistic fiber tracking was restricted using the white matter as a binary tissue classifier. Streamlines were included in the tractogram only if both endpoints of the streamline were in the gray matter mask. We used a Multi-Tissue Constrained Deconvolution model, with spherical harmonic order 8, to estimate an fiber orientation distribution (FOD) at each voxel [24]. These FODs were used, with a 30 degree angular threshold, to generate probabilistic tracking directions.

Using the whole brain tractograms, we identified the white matter pathways that were most promising for further analysis. For this study, we estimate a white matter pathway to be a path along all the streamlines that connect two gray matter regions. Using cortical and subcortical gray matter regions from the freesurfer segmentations, we identified 1102 such white matter pathways which were identified in all the subjects of our study. For each white matter pathway, we measured the average MD and FA to produce 2204 total measurements that we then used as features in our statistical analysis.

### 5.1.2 Statistics

For statistical analysis we used scipy and scickits learn [61, 62, 63]. Tissue volumes, cortical thickness, age, gender and BMI information were obtained from the HCP data set. We used a partial least squares (PLS) regression [64] to model the association between white matter measures, cortical thicknesses and BMI. In order to validate our results, we split the original dataset into two groups. The training group consisted of all 437 subjects released by the HCP in the S500 release or before. We set aside 329 subjects from the S900 release to be our test set. To determine the optimal number of latent variables for the PLS regression, we used a random 10 fold cross validation with 1000 iterations and fit models with

	slope	R	p
age	0.14	0.1	0.0051
gender	-0.52	-0.05	0.16
total GM volume	4.6e-07	0.0059	0.87
total WM volume	4.2e-06	0.046	0.2
total cortical GM volume	5.8e-07	0.0061	0.87
total sub-cortical GM volume	2.2e-05	0.024	0.51
fractional anisotropy	-27	-0.083	0.021
mean diffusivity	-5.6e+04	-0.17	2.4e-06

Table 5.1: This table shows the result of predicting BMI with each of the regressor independently using a linear regression. Gender, total GM and WM volumes as well as cortical and sub-cortical GM volumes were not correlated with BMI in our population. Age, FA, and MD showed a small, but statistically significant association with BMI.

up to 12 latent variables. Once the number of latent variables was established, we fit a PLS regression using all subjects from the training set and used the test set to validate the result. The results of the PLS regression were expressed as an image, denoted here as the effect map. See Appendix 1 for more on how the effect map was created. Using all the subjects in our training set, we warped the effect for each subject to MNI space and averaged the effect maps to produce a normalized, average effect map.

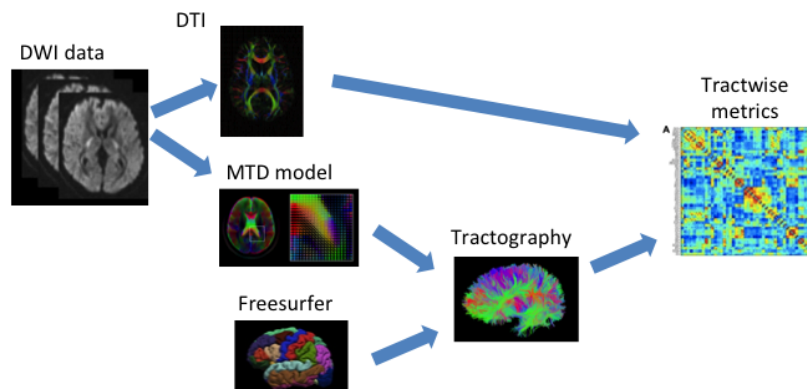


Figure 5.1: Tract based diffusion metrics are estimated by estimating individual WM pathways in each subject and extracting DTI metrics for those pathways.

## 5.2 Results

In our initial analysis we checked to see whether age, gender, total gray and white matter volume, cortical and sub-cortical gray matter volume, average fractional anisotropy, and



	t value	p value
age	1.1	0.27
fractional anisotropy	-4.6	5.5e-06
mean diffusivity	-5.9	4.2e-09

Table 5.2: A linear model for BMI using age, fractional anisotropy and mean diffusivity as features. The  $R^2$  value for this model was .06. When these features are modeled together, the association between age and BMI is no longer statistically significant.

average mean diffusivity were good independent predictors of BMI. Table 5.1 shows the results of this analysis. However, of the features we checked, only age, fractional anisotropy and mean diffusivity showed a statistically significant association with BMI. We fit a linear model using these three significant features, and found that FA and MD accounted for about 6% of the observed variance in BMI, however age no longer showed a statistically significant association with BMI when modeled with FA and MD. These results are summarized in Table 5.2. We also check to see the cortical thicknesses of the freesurfer defined cortical regions were good predictors of BMI. Table 5.4 shows the association between regional thickness and age adjusted BMI. We found that a significant number of regions in the frontal lobe, parietal lobe and anterior cingulate cortex showed a positive association with BMI. Also, three temporal lobe regions showed a strong negative association with BMI.

To further understand the associations between specific GM and WM regions and BMI we used a PLS regression model. PLS regression allows us to build a model with a large number of predictors, which may in themselves be largely correlated, without over-fitting the model. We fit PLS regression models with age, tract based diffusion metrics, and cortical thickness as predictors, five models in total. Table 5.3 shows how well each of these models predicted BMI in our data set. The number of latent variables for the PLS model were determined and the model was fit using the training data set (all subjects in the S500 and previous releases). The  $R^2$  values for each of the models was determined using the testing data set (new subjects in the S900 release). The models with tract based diffusion metrics included in the predictors did substantially better than the models that did not. The best model without diffusion metrics included age and GM thicknesses as predictors, and this model had an  $R^2$  of .045. The model with only tract based diffusion metrics had an  $R^2$  of .2. We did not find evidence that including cortical thicknesses or age in the model improved predictive performance above only using tract based diffusion metrics.

Figures 5.2 and 5.3 show a spacial representation of the PLS regression model for pre-

features used	latent variables	R <sup>2</sup>	p value
cortical thicknesses	2	0.024	0.0012
cortical thicknesses & age	3	0.045	0.0003
tract diffusion metrics	6	0.2	<1e-4
tract diffusion metrics & age	6	0.19	<1e-4
thicknesses & diffusion metrics	6	0.18	<1e-4

Table 5.3: How well cortical and WM features fit the data using the PLS regression model.

dicting BMI from tract based diffusion metrics. These spacial maps are an average of normalized effect maps across subjects. The effect maps show how much each tract or WM region contributes to the predicted BMI produced by the PLS model. Appendix 1 gives a formal definition of the effect map and more detail about how the effect maps are computed. Figure 5.2 shows the relative importance of FA values in the PLS model and Figure 5.3 similarly shows the relative importance of MD values. In the maps, red indicate a positive association with BMI and blue areas a negative association, however it is important to remember that these associations are relative to other regions. That is to say, the relative values within a subject are more important than the absolute FA and MD values in specific regions or tract. For example, the MD in the frontal aspect of the cingulate bundle has a large positive association with BMI while the MD in the temporal aspect of the same bundle has a large negative association. This pattern in the effect map tells us that the contrast between frontal and temporal MD in the cingulate bundle drives the BMI prediction more than the average MD or FA value in that bundle.

### 5.3 Discussion

Our study did not find an association between lower brain volume and higher BMI values in these young adults. Previous work in older populations has found that BMI is associated with reduced GM in many brain regions including the parietal, frontal occipital, and temporal lobes [65, 54]. However, in this study we only observed cortical thinning in three temporal GM regions. This apparent difference may be explained by the fact that the HCP subjects are much younger than the populations in these previous studies. This difference would arise if the reduced GM tissue observed in these studies of older individuals is due to increased aging related atrophy in persons with higher BMI. The aging hypothesis is consistent with previous work which found that dietary changes in young, obese mice can reverse obesity related cognitive declines [66]. Previous work has also shown that rats fed a high fat diet have higher levels pro-inflammatory cytokine expression in several brain

Cortical Area	R	p value	Cortical Area	R	p value
L Frontalpole	0.033	–	R Frontalpole	-0.0067	–
<b>L Medialorbitofrontal</b>	0.079	0.028	<b>R Medialorbitofrontal</b>	0.1	0.0057
L Lateralorbitofrontal	0.047	–	R Lateralorbitofrontal	0.071	–
L Parsorbitalis	0.032	–	R Parsorbitalis	0.025	–
L Parstriangularis	0.037	–	R Parstriangularis	0.045	–
L Parsopercularis	0.059	–	R Parsopercularis	0.042	–
<b>L Rostralmiddlefrontal</b>	0.098	0.0064	R Rostralmiddlefrontal	0.055	–
<b>L Caudalmiddlefrontal</b>	0.11	0.0027	R Caudalmiddlefrontal	0.054	–
<b>L Superiorfrontal</b>	0.078	0.031	R Superiorfrontal	0.061	–
<b>L Precentral</b>	0.086	0.017	R Precentral	0.052	–
L Paracentral	0.024	–	R Paracentral	0.032	–
L Insula	0.035	–	<b>R Insula</b>	0.075	0.038
<b>L Rostralanteriorcingulate</b>	0.18	4.1e-07	<b>R Rostralanteriorcingulate</b>	0.12	0.0012
<b>L Caudalanteriorcingulate</b>	0.08	0.028	R Caudalanteriorcingulate	-0.0093	–
L Posteriorcingulate	0.026	–	R Posteriorcingulate	-0.03	–
L Isthmuscingulate	0.04	–	R Isthmuscingulate	-0.043	–
L Entorhinal	-0.066	–	<b>R Entorhinal</b>	-0.14	6.5e-05
L Fusiform	0.063	–	R Fusiform	0.024	–
L Parahippocampal	0.041	–	R Parahippocampal	-0.0035	–
<b>L Inferiortemporal</b>	0.08	0.028	R Inferiortemporal	0.031	–
<b>L Temporalpole</b>	-0.13	0.00024	<b>R Temporalpole</b>	-0.12	0.00092
L Middletemporal	0.021	–	R Middletemporal	0.041	–
L Superiortemporal	0.015	–	R Superiortemporal	0.011	–
L Transversetemporal	-0.019	–	R Transversetemporal	-0.0019	–
L Bankssts	0.036	–	R Bankssts	0.015	–
L Postcentral	0.03	–	R Postcentral	0.049	–
L Supramarginal	0.046	–	<b>R Supramarginal</b>	0.087	0.016
<b>L Inferiorparietal</b>	0.087	0.016	<b>R Inferiorparietal</b>	0.1	0.0045
<b>L Superiorparietal</b>	0.11	0.0027	<b>R Superiorparietal</b>	0.15	2.8e-05
L Precuneus	0.043	–	R Precuneus	0.063	–
L Cuneus	-0.0012	–	R Cuneus	0.034	–
L Lateraloccipital	0.0047	–	R Lateraloccipital	0.054	–
L Pericalcarine	0.0057	–	R Pericalcarine	0.0092	–
L Lingual	0.047	–	R Lingual	0.03	–

Table 5.4: Association between cortical thicknesses and BMI. Some temporal regions show a negative association with BMI while frontal and parietal regions show a positive association with BMI.

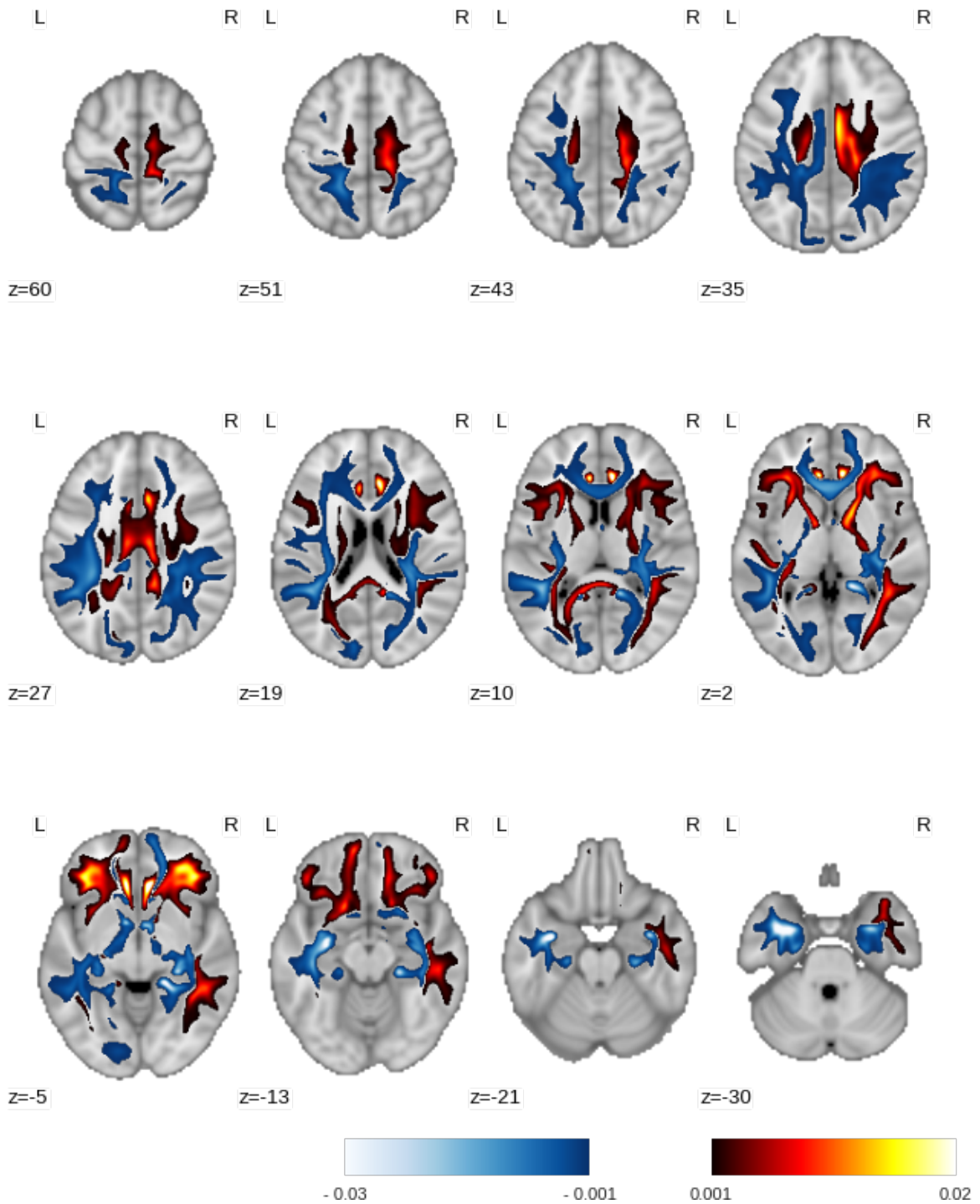


Figure 5.2: A spatial representation of the PLS model for predicting BMI. Higher FAs in red areas are associated with higher BMI and higher FAs in blue areas are associated with lower BMIs.

regions [67]. An inflammatory mechanism may be the reason that we observed a signifi-

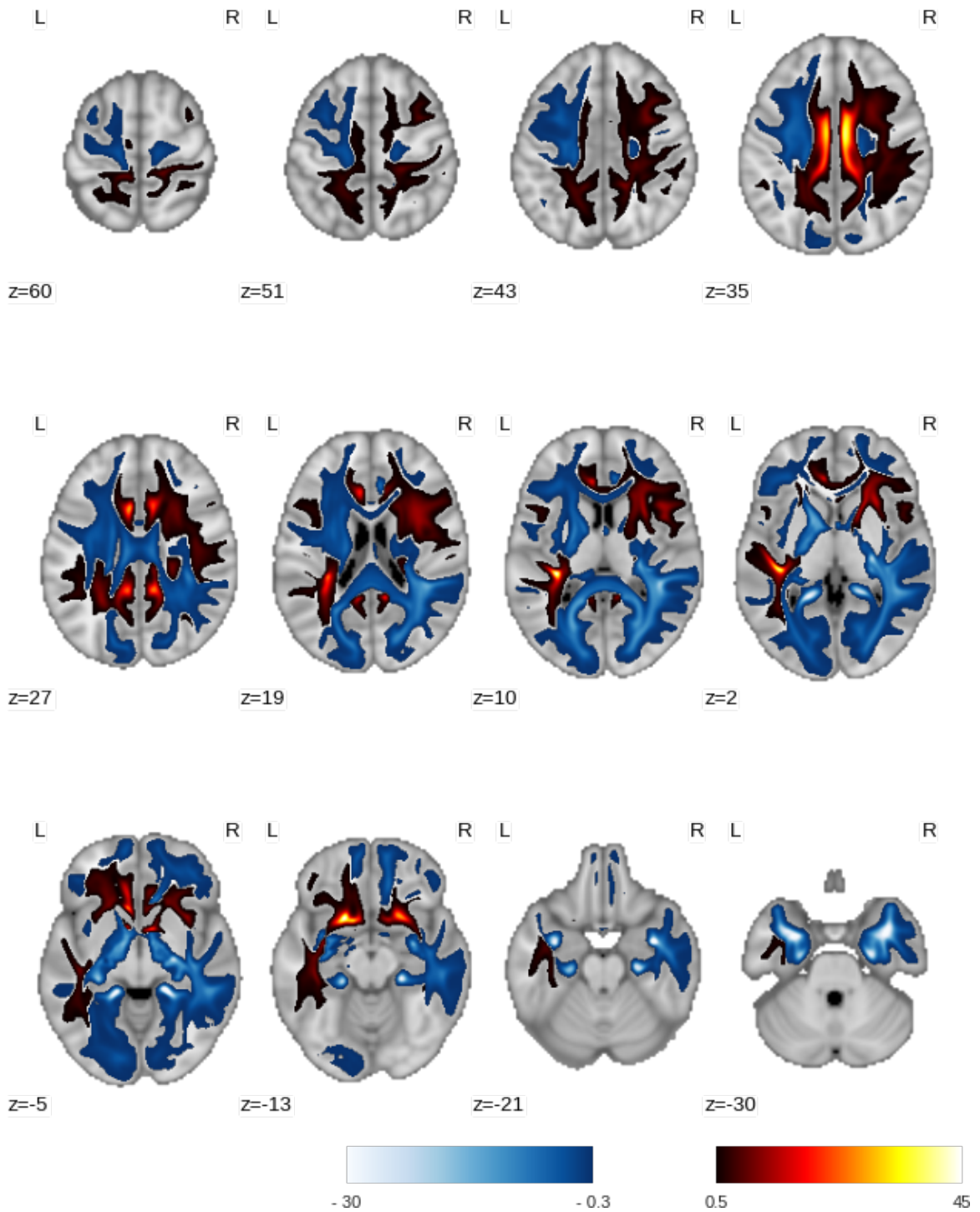


Figure 5.3: A spatial representation of the PLS model for predicting BMI. Higher MDs in red areas are associated with higher BMI and higher MDs in blue areas are associated with lower BMIs.

cant number of cortical GM regions which exhibit a positive association between cortical thickness and BMI. If both inflammation and atrophy contribute to the observed GM tissue thicknesses and volumes, the net effect may be small, at least in younger patients.

The strongest association we found in this population between imaging metrics and BMI was in the WM. Using a PLS model, the whole brain tract based diffusion metrics explain about 20% of the observed variance of BMI in this population. By contrast, cortical GM thicknesses and age explain less than 5% of that variance. Using our tractography approach, we modeled the white matter of the brain as a collection of a large number of overlapping WM pathways, both short range and long range, that when taken together constitute the connectivity of the brain. These pathways consist of both large bundles that mainly overlap with well known WM structures, such as the corpus callosum or superior lateral fasciculus, as well as smaller bundles that only minimally overlap with these well studied structures. This approach allowed us to organize regional WM information in a biologically motivated, data driven way without over simplifying the WM anatomy of the human brain. While tractography can be very noisy, the HCP data set has high resolution, multi shell diffusion MRI images with 270 unique gradient directions. The quality of this data, when combined with advanced diffusion modeling techniques, allowed us to create a robust and detailed model of the WM connectivity for each subject. Because the pathways of this connectivity model overlap, each voxel of the image can contribute to multiple pathways. When we used PLS regression to associate these pathways with BMI, we did not identify individual pathways that were specifically predictive of BMI, however when we computed effect maps from the PLS regression, we did find WM regions that contributed more strongly to the final BMI prediction.

The largest contributions to the PLS model were from regions associated with the cingulate bundle and uncinate fasciculus. Both these tracts connect frontal and temporal regions of the brain. We found that the FA and MD values in the temporal aspects of both these tracts contributed negatively to the BMI prediction. Additionally, the FA and MD of the anterior aspect of the cingulate bundle and the FA of some frontal WM areas contribute positively to the BMI prediction. To understand this relationship further, we computed the weighted average FA and MD in the red (positive contributions) and the blue (negative contributes) regions of the effect maps. We found that the average MD in both areas, red and blue, had a negative correlation with BMI. Also, the average FA in the blue regions had a negative correlation with BMI, but the FA in the red regions had no significant correlation. These findings are consistent with the linear model of BMI, which found a negative correlation with both whole brain WM average FA and MD relative to BMI. Taken together, these results tell us that higher BMI is associated with lower FA and MD in much of the

brain. This effect is strongest in the blue areas of the respective effect maps. However, higher FA and MD in the red regions of the respective effect maps is also predictive of higher BMI. This relationship exists either because the red regions are the least affected but higher BMI or because these red regions are the most correlated the blue regions in subjects with lower BMI and therefore serve as the best control regions for detecting BMI related differences in those blue regions. Both these explanations may account for some of the observed association in the model and the final result is that differences in FA and MD between the red and blue regions of the respective effect maps are more predictive of BMI than the FA and MD values taken from any given region of the brain.

Pervious studies have tried to identify WM effects related to BMI or obesity, but their results have been inconsistent. Our findings confirm some of the previously reported results, but may also help explain some of the inconsistency. Previous studies have found lower FA in some of the same WM pathways described in this paper. Specifically in studies of older subjects, one study reported lower FAs in the several parts of the corpus callosum, fornix, and cingulate bundle but did not find a relationship with FA in the uncinate fasciculus [56]. Another study, with similar methodology, considered these four WM tracts, but only found a relationship in the uncinate fasciculus, and no relationship between FA and BMI in the other three tracts [68]. The relationship between FA and BMI has also been studied in younger subjects, but with smaller sample sizes. One such study found lower AD (consistent with lower FA and lower MD) in the superior lateral fasciculus and anterior thalamic radiations while another study found lower FA in the mid-brain, internal capsule and perihippocampal white matter [69, 70]. The discrepancy in these results may be due to a lack of sensitivity in the methodologies used for these studies. DTI metrics exhibit a large intersubject and regional variability within subjects. Differences in how ROIs are defined in different subjects and in different studies can also contribute to estimation errors. If BMI is associated with DTI metric changes in much of the brain, as we have observed in this study, then small associations with BMI can be observed in many different regions. However, these observations will be limited by the sensitivity of the methodology and sample size used. By modeling the whole brain WM based on a connectivity model, we were able to measure a stronger association with BMI than any previously reported association.

## 5.4 Conclusion

In this study we used whole brain tractography to model the relationship between WM DTI metrics and BMI. We found that both FA and MD reductions were associated with higher BMI, especially in the temporal aspects of the cingulate bundle and uncinate fasciculus.

These associations were strongest when relative differences in MD and FA within each subject were accounted for. Three cortical regions in the temporal lobe also showed thinning associated with higher BMI.

## 5.5 Appendix

### 5.5.1 Effect Map

In this work we showed that BMI is associated with tract-wise white matter metrics and modeled that association using PLS regression. The PLS regression model can be projected back into image space in order to create an effect map. The effect map is a spacial representation of the PLS model and it can help unearth anatomical patterns in the white matter BMI association. This section defines the effect map.

The PLS regression model can be used to predict an outcome variable  $y_s$ , for subject  $s$ , from a feature vectors  $[x_{0s}, x_{1s}, \dots, x_{Ns}]$  using the following equation. In this equation  $L$  is the y loading and  $R$  is the x rotation matrix estimated by the PLS model. Also  $\bar{y}$  and  $\bar{x}_i$  are the population means of  $y_s$  and  $x_{is}$  respectively. Similarly,  $\sigma_y$  and  $\sigma_{x_i}$  are the standard deviations  $y_s$  and  $x_{is}$  respectively. These means and standard deviations are also estimated by the PLS model.

$$\hat{y}_s = \sigma_y \sum_c \sum_i L_c R_{ic} \frac{x_{is} - \bar{x}_i}{\sigma_{x_i}} + \bar{y} \quad (5.1)$$

By switching the order of the summations we can consolidate some of the terms and simplify the equation.

$$\hat{y}_s = \sum_i \frac{\sigma_y}{\sigma_{x_i}} \sum_c L_c R_{ic} (x_{is} - \bar{x}_i) + \bar{y} \quad (5.2)$$

We can combine the contributes the rotation and loading terms of the PLS model for each feature  $i$  into a single model coefficient  $a_i$ .

$$a_i = \frac{\sigma_y}{\sigma_{x_i}} \sum_c L_c R_{ic} \quad (5.3)$$

Now, we can represent  $\hat{y}$  in terms of these model coefficients.

$$\hat{y}_s = \sum_i a_i (x_{is} - \bar{x}_i) + \bar{y} \quad (5.4)$$

We can define also a constant  $k$  to consolidate the terms with no dependence on  $x_{is}$ .



$$k = \bar{y} - \sum_i a_i \bar{x}_i \quad (5.5)$$

$$\hat{y}_s = \sum_i a_i x_{is} + k \quad (5.6)$$

We have simplified the PLS prediction equation to a form that only requires a set of coefficients,  $[a_0, a_1, \dots]$  and a constant  $k$  to predict BMI from tract based metrics. We can now expand the features because each feature is the average value of a metric, MD for example, along a white matter tract in the brain. Each feature  $x_{is}$  can be expressed as a sum over  $m_{sv}$ , the value of the metric at voxel  $v$ , and  $d_{isv}$ , the density tract  $i$  at voxel  $v$ .

$$x_{is} = \frac{\sum_v d_{isv} m_{sv}}{\sum_v d_{isv}} \quad (5.7)$$

Using the expression for  $x_{is}$  in equation 5.7, we can express the prediction for  $\hat{y}$  as a sum over the voxels of an image.

$$\hat{y}_s = \sum_i a_i \frac{\sum_v d_{isv} m_{sv}}{\sum_v d_{isv}} + k \quad (5.8)$$

If we change the order of summation in equation 5.8, we get the following.

$$\hat{y}_s = \sum_v \frac{\sum_i a_i d_{isv}}{\sum_v d_{isv}} m_{sv} + k \quad (5.9)$$

From equation 5.9 we can consolidate terms into an effect value  $E_{sv}$ . The effect value for each voxel  $v$  is the a sum of the model coefficients for each tract associated with that voxel, weighted by how much that voxel contributes to that tract.

$$E_{sv} = \frac{\sum_i a_i d_{isv}}{\sum_v d_{isv}} \quad (5.10)$$

The prediction of  $\hat{y}_s$  can now be expressed in terms of the effect values and metric map.

$$\hat{y}_s = \sum_v E_{sv} m_{sv} + k \quad (5.11)$$

When represented as an image, we refer the effect values  $E_{sv}$  as an effect map. The effect map is in essence a set of weights which, when combined with a metric map, can predict the outcome variable for a subject  $s$ . Each subject has a unique effect map because the effect map is derived from the tract densities,  $d_{isv}$ , specific to that subject. To compute a population average effect, we normalized the effect maps of each subject into a standard

space, MNI space, and averaged across subjects.

# Bibliography

- [1] Mehdi Poustchi-Amin, Scott A. Mirowitz, Jeffrey J. Brown, Robert C. McKinstry, and Tao Li. Principles and Applications of Echo-planar Imaging: A Review for the General Radiologist. *RadioGraphics*, 21(3):767–779, may 2001.
- [2] R.J. Ordidge, P. Mansfield, M. Doyle, and R.E. Coupland. Real-time movie images by nuclear magnetic resonance. *Magnetic Resonance Imaging*, 2(2):150, jan 1984.
- [3] David A. Feinberg and Koichi Oshio. Phase errors in multi-shot echo planar imaging. *Magnetic Resonance in Medicine*, 32(4):535–539, oct 1994.
- [4] Magnetic Resonance Imaging. In *Magnetic Resonance Imaging*, pages 1–17. Wiley-Blackwell, may 2014.
- [5] Pratik Mukherjee, Mark M. Bahn, Robert C. McKinstry, Joshua S. Shimony, Thomas S. Cull, Erbil Akbudak, Abraham Z. Snyder, and Thomas E. Conturo. Differences between Gray Matter and White Matter Water Diffusion in Stroke: Diffusion-Tensor MR Imaging in 12 Patients. *Radiology*, 215(1):211–220, apr 2000.
- [6] Khader M. Hasan, Dennis L. Parker, and Andrew L. Alexander. Comparison of gradient encoding schemes for diffusion-tensor MRI. *Journal of Magnetic Resonance Imaging*, 13(5):769–780, 2001.
- [7] Derek K. Jones. The effect of gradient sampling schemes on measures derived from diffusion tensor MRI: A Monte Carlo study. *Magnetic Resonance in Medicine*, 51(4):807–815, 2004.
- [8] Chris A. Clark and Denis Le Bihan. Water diffusion compartmentation and anisotropy at high b values in the human brain. *Magnetic Resonance in Medicine*, 44(6):852–859, 2000.

- [9] Marta Morgado Correia, Thomas A. Carpenter, and Guy B. Williams. Looking for the optimal DTI acquisition scheme given a maximum scan time: are more b-values a waste of time? *Magnetic Resonance Imaging*, 27(2):163–175, feb 2009.
- [10] Derek K. Jones and Mara Cercignani. Twenty-five pitfalls in the analysis of diffusion MRI data. *NMR in Biomedicine*, 23(7):803–820, sep 2010.
- [11] Lin-Ching Chang, Derek K. Jones, and Carlo Pierpaoli. RESTORE: Robust estimation of tensors by outlier rejection. *Magnetic Resonance in Medicine*, 53(5):1088–1095, 2005.
- [12] David S. Tuch, Timothy G. Reese, Mette R. Wiegell, Nikos Makris, John W. Belliveau, and Van J. Wedeen. High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. *Magnetic Resonance in Medicine*, 48(4):577–582, sep 2002.
- [13] Van J. Wedeen, Patric Hagmann, Wen-Yih Isaac Tseng, Timothy G. Reese, and Robert M. Weisskoff. Mapping complex tissue architecture with diffusion spectrum magnetic resonance imaging. *Magnetic Resonance in Medicine*, 54(6):1377–1386, 2005.
- [14] V.J. Wedeen, R.P. Wang, J.D. Schmahmann, T. Benner, W.Y.I. Tseng, G. Dai, D.N. Pandya, P. Hagmann, H. D'Arceuil, and A.J. de Crespigny. Diffusion spectrum magnetic resonance imaging (DSI) tractography of crossing fibers. *NeuroImage*, 41(4):1267–1277, jul 2008.
- [15] David S. Tuch. Q-ball imaging. *Magnetic Resonance in Medicine*, 52(6):1358–1372, 2004.
- [16] Iman Aganj, Christophe Lenglet, Guillermo Sapiro, Essa Yacoub, Kamil Ugurbil, and Noam Harel. Reconstruction of the orientation distribution function in single- and multiple-shell q-ball imaging within constant solid angle. *Magnetic Resonance in Medicine*, pages n/a–n/a, jun 2010.
- [17] Antonio Tristán-Vega, Carl-Fredrik Westin, and Santiago Aja-Fernández. Estimation of fiber Orientation Probability Density Functions in High Angular Resolution Diffusion Imaging. *NeuroImage*, 47(2):638–650, aug 2009.
- [18] David S. Tuch, Timothy G. Reese, Mette R. Wiegell, and Van J. Wedeen. Diffusion MRI of Complex Neural Architecture. *Neuron*, 40(5):885–895, dec 2003.

- [19] Christopher P. Hess, Pratik Mukherjee, Eric T. Han, Duan Xu, and Daniel B. Vigneron. Q-ball reconstruction of multimodal fiber orientations using the spherical harmonic basis. *Magnetic Resonance in Medicine*, 56(1):104–117, 2006.
- [20] Maxime Descoteaux, Elaine Angelino, Shaun Fitzgibbons, and Rachid Deriche. Regularized fast, and robust analytical Q-ball imaging. *Magnetic Resonance in Medicine*, 58(3):497–510, 2007.
- [21] Chantal M.W. Tax, Ben Jeurissen, Sjoerd B. Vos, Max A. Viergever, and Alexander Leemans. Recursive calibration of the fiber response function for spherical deconvolution of diffusion MRI data. *NeuroImage*, 86:67–80, feb 2014.
- [22] J.-Donald Tournier, Fernando Calamante, David G. Gadian, and Alan Connelly. Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution. *NeuroImage*, 23(3):1176–1185, nov 2004.
- [23] J-Donald Tournier, Fernando Calamante, and Alan Connelly. Robust determination of the fibre orientation distribution in diffusion MRI: Non-negativity constrained super-resolved spherical deconvolution. *NeuroImage*, 35(4):1459–1472, may 2007.
- [24] Ben Jeurissen, Jacques-Donald Tournier, Thijs Dhollander, Alan Connelly, and Jan Sijbers. Multi-tissue constrained spherical deconvolution for improved analysis of multi-shell diffusion MRI data. *NeuroImage*, 103:411–426, dec 2014.
- [25] Susumu Mori, Barbara J. Crain, V. P. Chacko, and Peter C. M. Van Zijl. Three-dimensional tracking of axonal projections in the brain by magnetic resonance imaging. *Annals of Neurology*, 45(2):265–269, feb 1999.
- [26] Philip M. Robson, Aaron K. Grant, Ananth J. Madhuranthakam, Riccardo Lattanzi, Daniel K. Sodickson, and Charles A. McKenzie. Comprehensive quantification of signal-to-noise ratio and g-factor for image-based and k-space-based parallel imaging reconstructions. *Magnetic Resonance in Medicine*, 60(4):895–907, oct 2008.
- [27] Peter Kellman and Elliot R. McVeigh. Image reconstruction in SNR units: A general method for SNR measurement. *Magnetic Resonance in Medicine*, 54(6):1439–1447, 2005.
- [28] CG Koay, E Ozarslan, and C Pierpaoli. Probabilistic Identification and Estimation of Noise (PIESNO): a self-consistent approach and its applications in MRI. *J Magn Reson*, 199:94–103, Jul 2009.

- [29] T Pieciak, S Aja-Fernandez, and Sanchez-Ferrero G Vegas. Non-Stationary Rician Noise Estimation in Parallel MRI using a Single Image: a Variance-Stabilizing Approach. *IEEE Trans Pattern Anal Mach Intell*, Nov 2016.
- [30] Olaf Dietrich, José G. Raya, Scott B. Reeder, Maximilian F. Reiser, and Stefan O. Schoenberg. Measurement of signal-to-noise ratios in MR images: Influence of multichannel coils parallel imaging, and reconstruction filters. *Journal of Magnetic Resonance Imaging*, 26(2):375–385, 2007.
- [31] Eleftherios Garyfallidis, Matthew Brett, Bagrat Amirbekian, Ariel Rokem, Stefan van der Walt, Maxime Descoteaux, and Ian Nimmo-Smith. Dipy a library for the analysis of diffusion MRI data. *Front. Neuroinform.*, 8, feb 2014.
- [32] SungWon Chung, Ying Lu, and Roland G. Henry. Comparison of bootstrap approaches for estimation of uncertainties of DTI parameters. *NeuroImage*, 33(2):531–541, nov 2006.
- [33] H.A. Haroon, D.M. Morris, K.V. Embleton, D.C. Alexander, and G. Parker. Using the Model-Based Residual Bootstrap to Quantify Uncertainty in Fiber Orientations From formula formulatype=inline tex Notation=TeX\$Q\$/tex/formula-Ball Analysis. *IEEE Transactions on Medical Imaging*, 28(4):535–550, apr 2009.
- [34] JI Berman, S Chung, P Mukherjee, CP Hess, ET Han, and RG Henry. Probabilistic streamline q-ball tractography using the residual bootstrap. *Neuroimage*, 39:215–22, Jan 2008.
- [35] Jelle Veraart, Jan Sijbers, Stefan Sunaert, Alexander Leemans, and Ben Jeurissen. Weighted linear least squares estimation of diffusion MRI parameters: Strengths limitations, and pitfalls. *NeuroImage*, 81:335–346, nov 2013.
- [36] P. Coupe, P. Yger, S. Prima, P. Hellier, C. Kervrann, and C. Barillot. An Optimized Blockwise Nonlocal Means Denoising Filter for 3-D Magnetic Resonance Images. *IEEE Transactions on Medical Imaging*, 27(4):425–441, apr 2008.
- [37] A.C. Davison and D.V. Hinkley. *Bootstrap Methods and Their Application*. Cambridge Series in Statistical and Probabilistic Mathematics. Cambridge University Press, 1997.
- [38] Hui Zhang, Torben Schneider, Claudia A. Wheeler-Kingshott, and Daniel C. Alexander. NODDI: Practical in vivo neurite orientation dispersion and density imaging of the human brain. *NeuroImage*, 61(4):1000–1016, jul 2012.

- [39] Yaniv Assaf and Peter J. Basser. Composite hindered and restricted model of diffusion (CHARMED) MR imaging of the human brain. *NeuroImage*, 27(1):48–58, aug 2005.
- [40] Anders M. Dale, Bruce Fischl, and Martin I. Sereno. Cortical Surface-Based Analysis. *NeuroImage*, 9(2):179–194, feb 1999.
- [41] Y. Zhang, M. Brady, and S. Smith. Segmentation of brain MR images through a hidden Markov random field model and the expectation-maximization algorithm. *IEEE Transactions on Medical Imaging*, 20(1):45–57, 2001.
- [42] Robert E. Smith, Jacques-Donald Tournier, Fernando Calamante, and Alan Connelly. Anatomically-constrained tractography: Improved diffusion MRI streamlines tractography through effective use of anatomical information. *NeuroImage*, 62(3):1924–1938, sep 2012.
- [43] Gabriel Girard, Kevin Whittingstall, Rachid Deriche, and Maxime Descoteaux. Towards quantitative connectivity analysis: reducing tractography biases. *Neuroimage*, 98:266–278, 2014.
- [44] Fernando Calamante, Jacques-Donald Tournier, Graeme D. Jackson, and Alan Connelly. Track-density imaging (TDI): Super-resolution white matter imaging using whole-brain track-density mapping. *NeuroImage*, 53(4):1233–1243, dec 2010.
- [45] F. Calamante. Super-Resolution Track Density Imaging: Anatomic Detail versus Quantification. *American Journal of Neuroradiology*, 37(6):1066–1067, feb 2016.
- [46] Molly Memel, Kyle Bourassa, Cindy Woolverton, and David A. Sbarra. Body Mass and Physical Activity Uniquely Predict Change in Cognition for Aging Adults. *Annals of Behavioral Medicine*, 50(3):397–408, jan 2016.
- [47] P. L. Yau, M. G. Castro, A. Tagani, W. H. Tsui, and A. Convit. Obesity and Metabolic Syndrome and Functional and Structural Brain Impairments in Adolescence. *PEDI-ATRICES*, 130(4):e856–e864, sep 2012.
- [48] John Gunstad, Robert H. Paul, Ronald A. Cohen, David F. Tate, Mary Beth Spitznagel, and Evian Gordon. Elevated body mass index is associated with executive dysfunction in otherwise healthy adults. *Comprehensive Psychiatry*, 48(1):57–61, jan 2007.
- [49] Jason C. D. Nguyen, A. Simon Killcross, and Trisha A. Jenkins. Obesity and cognitive decline: role of inflammation and vascular changes. *Frontiers in Neuroscience*, 8, nov 2014.

- [50] J Liang, B E Matheson, W H Kaye, and K N Boutelle. Neurocognitive correlates of obesity and obesity-related behaviors in children and adolescents. *Int J Obes Relat Metab Disord*, 38(4):494–506, aug 2013.
- [51] Andrew Fedor and John Gunstad. Higher BMI Is Associated with Reduced Cognitive Performance in Division I Athletes. *Obesity Facts*, 6(2):185–192, 2013.
- [52] Chun Liang Hsu, Michelle W. Voss, John R. Best, Todd C. Handy, Kenneth Madden, Niousha Bolandzadeh, and Teresa Liu-Ambrose. Elevated body mass index and maintenance of cognitive function in late life: exploring underlying neural mechanisms. *Frontiers in Aging Neuroscience*, 7, aug 2015.
- [53] April J. Ho, Cyrus A. Raji, James T. Becker, Oscar L. Lopez, Lewis H. Kuller, Xue Hua, Suh Lee, Derrek Hibar, Ivo D. Dinov, Jason L. Stein, Clifford R. Jack, Michael W. Weiner, Arthur W. Toga, and Paul M. Thompson. Obesity is linked with lower brain volume in 700 AD and MCI patients. *Neurobiology of Aging*, 31(8):1326–1339, aug 2010.
- [54] Cyrus A. Raji, April J. Ho, Neelroop N. Parikshak, James T. Becker, Oscar L. Lopez, Lewis H. Kuller, Xue Hua, Alex D. Leow, Arthur W. Toga, and Paul M. Thompson. Brain structure and obesity. *Human Brain Mapping*, pages NA–NA, 2009.
- [55] Auriel A. Willette and Dimitrios Kapogiannis. Does the brain shrink as the waist expands? *Ageing Research Reviews*, 20:86–97, mar 2015.
- [56] Brianne Bettcher, Christine Walsh, Christa Watson, Joshua Miller, Ralph Green, Nihar Patel, Bruce Miller, Kristine Yaffe, and Joel H. Kramer. Body mass and white matter integrity: The role of vascular and inflammatory factors in healthy aging. *Alzheimer's & Dementia*, 9(4):P392–P393, jul 2013.
- [57] Stefan Gazdzinski, John Kornak, Michael W. Weiner, and Dieter J. Meyerhoff. Body mass index and magnetic resonance markers of brain integrity in adults. *Annals of Neurology*, 63(5):652–657, may 2008.
- [58] David C. Van Essen, Stephen M. Smith, Deanna M. Barch, Timothy E.J. Behrens, Essa Yacoub, and Kamil Ugurbil. The WU-Minn Human Connectome Project: An overview. *NeuroImage*, 80:62–79, oct 2013.
- [59] Stamatios N. Sotiropoulos, Saad Jbabdi, Junqian Xu, Jesper L. Andersson, Steen Moeller, Edward J. Auerbach, Matthew F. Glasser, Moises Hernandez, Guillermo



- Sapiro, Mark Jenkinson, David A. Feinberg, Essa Yacoub, Christophe Lenglet, David C. Van Essen, Kamil Ugurbil, and Timothy E.J. Behrens. Advances in diffusion MRI acquisition and processing in the Human Connectome Project. *NeuroImage*, 80:125–143, oct 2013.
- [60] Matthew F. Glasser, Stamatios N. Sotiropoulos, J. Anthony Wilson, Timothy S. Coalson, Bruce Fischl, Jesper L. Andersson, Junqian Xu, Saad Jbabdi, Matthew Webster, Jonathan R. Polimeni, David C. Van Essen, and Mark Jenkinson. The minimal pre-processing pipelines for the Human Connectome Project. *NeuroImage*, 80:105–124, oct 2013.
- [61] Eric Jones, Travis Oliphant, P Peterson, et al. Open source scientific tools for Python, 2001.
- [62] Christian Hill. SciPy. In *Learning Scientific Programming with Python*, pages 333–401. Cambridge University Press (CUP).
- [63] Fabio Nelli. Machine Learning with scikit-learn. In *Python Data Analytics*, pages 237–264. Springer Science Business Media, 2015.
- [64] Adrian Leguina. A primer on partial least squares structural equation modeling (PLS-SEM). *International Journal of Research & Method in Education*, 38(2):220–221, 2015.
- [65] April J. Ho, Cyrus A. Raji, James T. Becker, Oscar L. Lopez, Lewis H. Kuller, Xue Hua, Ivo D. Dinov, Jason L. Stein, Caterina Rosano, Arthur W. Toga, and Paul M. Thompson. The effects of physical activity education, and body mass index on the aging brain. *Human Brain Mapping*, 32(9):1371–1382, aug 2010.
- [66] Catrina Sims-Robinson, Anna Bakeman, Elizabeth Bruno, Samuel Jackson, Rebecca Glasser, Geoffrey G. Murphy, and Eva L. Feldman. Dietary Reversal Ameliorates Short- and Long-Term Memory Deficits Induced by High-fat Diet Early in Life. *PLOS ONE*, 11(9):e0163883, sep 2016.
- [67] Chloé Boitard, Amandine Cavaroc, Julie Sauvart, Agnès Aubert, Nathalie Castanon, Sophie Layé, and Guillaume Ferreira. Impairment of hippocampal-dependent memory induced by juvenile high-fat diet intake is associated with enhanced hippocampal inflammation in rats. *Brain Behavior, and Immunity*, 40:9–17, aug 2014.
- [68] Jacob D. Bolzenius, David H. Laidlaw, Ryan P. Cabeen, Thomas E. Conturo, Amanda R. McMichael, Elizabeth M. Lane, Jodi M. Heaps, Lauren E. Salminen,

Laurie M. Baker, Staci E. Scott, Sarah A. Cooley, John Gunstad, and Robert H. Paul. Brain structure and cognitive correlates of body mass index in healthy older adults. *Behavioural Brain Research*, 278:342–347, feb 2015.

[69] Timothy D. Verstynen, Andrea M. Weinstein, Walter W. Schneider, John M. Jakicic, Dana L. Rofey, and Kirk I. Erickson. Increased Body Mass Index Is Associated With a Global and Distributed Decrease in White Matter Microstructural Integrity. *Psychosomatic Medicine*, 74(7):682–690, sep 2012.

[70] Stephanie Kullmann, Martina F. Callaghan, Martin Heni, Nikolaus Weiskopf, Klaus Scheffler, Hans-Ulrich Häring, Andreas Fritsche, Ralf Veit, and Hubert Preissl. Specific white matter tissue microstructure changes associated with obesity. *NeuroImage*, 125:36–44, jan 2016.

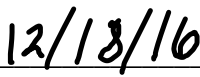
**Publishing Agreement**

*It is the policy of the University to encourage the distribution of all theses, dissertations, and manuscripts. Copies of all UCSF theses, dissertations, and manuscripts will be routed to the library via the Graduate Division. The library will make all theses, dissertations, and manuscripts accessible to the public and will preserve these to the best of their abilities, in perpetuity.*

***Please sign the following statement:***

*I hereby grant permission to the Graduate Division of the University of California, San Francisco to release copies of my thesis, dissertation, or manuscript to the Campus Library to provide access and preservation, in whole or in part, in perpetuity.*

  
\_\_\_\_\_  
Author Signature

  
\_\_\_\_\_  
Date