On some Complex and Massive Data Problems

By

Ka Wai Wong
B.Sc. (Chinese University of Hong Kong) 2008
M.Phil. (Chinese University of Hong Kong) 2010

Dissertation

Submitted in partial satisfaction of the requirements for the degree of

Doctor of Philosophy

in

Statistics

in the

Office of Graduate Studies

of the

University of California

Davis

Approved:

Thomas C. M. Lee, Chair

Debashis Paul

Jie Peng

Committee in Charge

2014

Copyright © 2014 by
Ka Wai Wong
All rights reserved.

For my loved ones.

For my stubbornness.

Contents

Abstract

On some Complex and Massive Data Problems

In this thesis, we develop statistical solutions to some complex and massive data problems, which involve modern data complications such as big data size, black-box structures, complex data acquisitions and manifold structures. These problems stem from astronomy, brain imaging and computer experiments. Novel, efficient and tailored statistical methodologies are developed to cope with the unique difficulties of each problem.

Acknowledgments

I would like to express my deepest gratitude to Prof. Thomas C. M. Lee, who supported me throughout my Ph.D. study with his encouragement, patience and knowledge while allowing room for my stubbornness. He provided me with valuable advice, practical research experience, inspiring ideas and most importantly, good company. I am also grateful to Prof. Alexander Aue, Prof. Paul Baines, Prof. Vinay L. Kashyap, Prof. Debashis Paul, Prof. Jie Peng and Dr. Curtis B. Storlie for their time, patience and constructive ideas. Without their help and guidance, I would not be able to finish most of the work in this thesis. I am also indebted to the faculty members, graduate students and the staff of the Department of Statistics, UC Davis, for providing a stimulating and fun environment.

Moreover, I want to express my thanks to my parents for their continuous support throughout my studies. Finally, and most importantly, a special thank goes to my wife, Amy Lam, for her constant support, understanding and encouragement.

Chapter 1

Overview

1.1 Introduction

In this thesis, statistical solutions to four different data problems are developed. These data problems involve various modern data complications such as big data size, black-box structures, complex data acquisitions and manifold structures. They stem from three different areas: astronomy, brain imaging and computer experiments, in which the corresponding data problems inherit the domain's unique difficulties. In the following, these four pieces of work are briefly described.

1.2 Brain imaging

The recent advancement of neuroimaging technology has generated a huge amount of brain imaging data. These images are not only large, but also complex. They carry the mission of understanding the incredibly complicated brain structures. The analysis of such data poses many challenging statistical problems that require both accurate modeling and fast algorithms.

In Chapter 2, we focus on diffusion magnetic resonance imaging (dMRI), which is one of the more recent medical imaging techniques. It is used to measure the diffusion of water molecules along a pre-specified set of gradient directions. The data consist of signals measured along these pre-specified gradient directions for each voxel in a three dimensional space. A fiber or fiber bundle lies across voxels and its direction

can be detected through these signals. The goal is to track these fibers. In our work, we investigate a non-identifiability issue of the commonly used diffusion tensor modeling and propose a novel route for solving the tracking problem. We also develop smoothing methods for fiber orientations under a mixture of orientation populations. Here, the space of fiber orientations is a special manifold called real projective space and this requires a non-standard smoothing technique. In our work, we show that the proposed method enjoys desirable asymptotic properties and produces excellent results to a dMRI dataset from Alzheimer's Disease Neuroimaging Initiative.

1.3 Computer experiments

In many areas, complex mathematical models are used to model the physical reality and these models are typically implemented as computer codes. A computer experiment is referred to as gathering data through running the computer codes with various values of input parameters. This strategy is widely used in areas where physical experiments are too expensive or simply impossible, but a reasonable mathematical model can be implemented as computer codes. However, even computer experiments (computer models) are relatively easier to conduct, these codes are still sophisticated and time costly to run due to the complexity of the underlying mathematical models. Thus, another layer of statistical modeling is usually assumed for the computer model to form a fast surrogate, which is called an emulator. In addition, uncertainty quantification is important in this area since the resources usually are limited, which prevents one from routinely getting data, and thus one would want have some idea about the uncertainty of the estimates.

In Chapter 3, we work on statistical approaches to incorporating uncertainty quantification (UQ) into global optimization techniques, which are intended to be used for optimizing high dimensional expensive black-box functions (e.g. computer models). Our work refines high dimensional optimization techniques through variable selection, sequential sampling and incorporation of uncertainty into the function estimation. We developed an efficient algorithm for construction of confidence sets in order

to quantify the uncertainty of the estimated optimum. This can serve as a guideline for determining when the estimated optimum is satisfactory and can be used to safeguard scenarios of multiple global optima or occurrences of competitive local optima.

In Chapter 4, we develop a frequentist framework for computer calibration problems, for which Bayesian methods are dominate. The common calibration framework involves a semi-parametric model which leads to identifiability issue. In our work, we provide an intuitive and identifiable parametrization for this semi-parametric model, which lay down a general computer calibration framework from a frequentist's angle. The flexibility of the framework allows practitioners to have their own choices of surrogates. Our approach enjoys desirable theoretical properties. In addition, we propose a bootstrapping approach for uncertainty quantification. The bootstrap is coherent with our proposed framework and together allows for flexibility for practitioners' choices of surrogates for the computer model, nonparametric models for the discrepancy function and methods of global optimization.

For both Chapter 3 and Chapter 4, since permissions have not yet been granted from the data providers to disclose the corresponding analysis, their corresponding real data applications are not presented in this thesis.

1.4 Astronomy

Due to technological breakthroughs, more and more powerful and complex telescopes have been set up. This leads to strong demands for statistical tools and the rise of a new interdisciplinary field called astrostatistics. Telescopic data are usually complex, involving many instrumental adjustments. The modeling of telescopic data are usually complicated and requires hierarchical structures, but fast computations are usually required for the increasing amount of astronomical data.

In Chapter 5 (Wong *et al.*, 2014), we study a broken-power law models for the well-known $\log(N)$-$\log(S)$ relationship in astrophysics community. The major challenge of using this model arises from the unusual difficulty in obtaining the maximum likelihood estimation of its parameters. In this work, we construct an efficient algo-

rithm through a newly proposed interwoven expectation-maximization strategy. This strategy combines powers of different data augmentation schemes to achieve fast and stable performances. In our work, we apply our method successfully to a Chandra Deep Field North dataset. This work has already been accepted for publication in the Annals of Applied Statistics.

1.5 Outline of the thesis

The following chapters (with the corresponding appendix) of this thesis are self-contained, with each of them focuses on a single problem. Chapter 2 is about the fiber estimation of diffusion MRI. As for Chapter 3 and Chapter 4, we focus on computer experiments. Global optimization of high dimensional expensive black-box systems and a frequentist approach to computer model calibration are discussed. In Chapter 5, the work about automatic estimation of flux distributions of astrophysical source populations is presented.

Chapter 2

Fiber Direction Estimation in Diffusion MRI

Abstract

Diffusion magnetic resonance imaging is a medical imaging technology to probe anatomical architectures of biological samples in an in vivo and non-invasive manner through measuring water diffusion. It is widely used to reconstruct white matter fiber tracts in brains. This can be done in several steps. Typically, the first step is to estimate the diffusion direction(s) for each voxel of the biological sample under study by extracting the leading eigenvector from the estimated diffusion tensor at each voxel. As it is reasonable to assume that the diffusion directions from neighboring voxels are similar, a local smoothing may be applied to the estimated tensors or directions to improve the estimation of diffusion directions. Finally, a tracking algorithm is used to reconstruct fiber tracts based on (estimated) diffusion directions.

Most commonly used tensor estimation methods assume a single tensor and do not work well when there are multiple principal diffusion directions within a single voxel. The first contribution of this paper is the proposal of a new method which is able to identify and estimate multiple diffusion directions within a voxel. This method is based on a new parametrization of the multi-tensor model and it produces reliable results even when there are multiple principal diffusion directions within the voxels. As a second contribution, this paper proposes a novel

direction smoothing method which greatly improves diffusion direction estimation in regions with crossing fibers. This smoothing method is shown to have excellent theoretical and empirical properties. Lastly this paper develops a novel fiber tracking algorithm which takes (estimated) diffusion directions as input and accommodates multiple directions within a voxel. The overall methodology is illustrated with data sets collected for the study of Alzheimer's disease.

This is a joint work with Thomas C. M. Lee[1], Debashis Paul[1] and Jie Peng[1].

2.1 Introduction

Diffusion magnetic resonance imaging (dMRI) is a medical imaging technology that uses magnetic field gradients to measure water diffusion on a three-dimensional (3D) grid of biological tissue along a set of predetermined directions (Bammer *et al.*, 2009; Beaulieu, 2002; Chanraud *et al.*, 2010; Mukherjee *et al.*, 2008). In biological tissues, water diffusion is anisotropic due to the presence of fiber bundles with coherent orientations and thus anatomical structures can be deduced from the diffusion characteristics of water. Due to its in vivo and non-invasive nature, dMRI has been widely applied to delineate the white matter fiber tracts in human brain. Mapping white matter fiber tracts is of great importance in the study of neuronal connectivity and understanding of brain functionality (Mori, 2007; Sporns, 2011).

Water diffusion in any location in the brain is often modeled as a 3D Gaussian process. At each voxel, diffusion is described by a 3×3 positive definite matrix, which is referred to as a diffusion tensor; see Mori (2007) for an introduction to diffusion tensor imaging (DTI) techniques. One then extracts the direction information from the estimated diffusion tensor (e.g., the principal eigenvector) at each voxel and reconstructs the white matter fiber tracts by computer aided tracking algorithms via a process named tractography (Basser *et al.*, 2000).

However, DTI cannot resolve multiple fiber populations with distinct orientations (i.e., crossing fibers) within a voxel since a tensor only has one principal direction.

[1]Department of Statistics, University of California at Davis

In crossing fiber regions, estimated diffusion tensors may lead to low anisotropy estimation or oblate tensor estimation. Poor tensor estimation results in poor direction estimation which affects fiber reconstruction, e.g., early termination of the fiber tracking or biased fiber tracking.

In order to resolve intravoxel orientational heterogeneity, several approaches have been proposed. Tuch *et al.* (2002) propose a multi-tensor model which assumes a finite number of homogeneous fiber directions with a voxel and Gaussian diffusion along each direction. However, it has been shown that the parameters in the multi-tensor model are not identifiable (Scherrer and Warfield, 2010). Nonparametric methods such as Q-ball and Q-space imaging have been proposed (Descoteaux *et al.*, 2007; Tuch, 2004). However such methods rely on high angular resolution diffusion imaging (HARDI) (Hosey *et al.*, 2005; Tuch *et al.*, 2002) where a large number of gradients is sampled (e.g., a few hundreds). Most currently available data sets, and particularly those obtained under clinical settings, have much less number of gradient directions (a few tens at most), rendering such methods not applicable.

The primary goal of this paper is to develop a new method for fiber detection and tracking that works exceptionally well in the presence of crossing fibers. Our method is completely automatic and improves existing methods in several aspects. Loosely, the method can be divided into the following three major steps.

In the first step, we estimate the tensor directions within each voxel under a multi-tensor model. We propose a new parametrization which makes the tensor directions identifiable. We develop an efficient and numerically stable computational procedure to obtain the global MLE of the tensor directions.

Once the tensor direction estimates are obtained for all individual voxels, in the second step, a direction smoothing procedure is applied to further improve the diffusion direction estimates by borrowing information from neighboring voxels. A distinctive and unique feature of this new procedure is that it handles crossing fibers through the clustering of directions into homogeneous groups. We note that, although various tensor smoothing methods have been proposed (e.g., Arsigny *et al.*,

2006; Carmichael *et al.*, 2013; Fillard *et al.*, 2007; Fletcher and Joshi, 2007; Pennec *et al.*, 2006; Yuan *et al.*, 2012), to the best of our knowledge, little work in the literature on direct diffusion direction smoothing. Since diffusion directions rather than tensors are used as input for tracking algorithms, methods on direction estimation and smoothing should be more efficient in terms of fiber tracking.

In the last step, a fiber tracking algorithm is applied to reconstruct fiber tracts through (smoothed) diffusion direction estimates. Our tracking algorithm is designed to explicitly allow for multiple directions within a voxel.

It is shown by extensive numerical studies that the proposed procedure is effective in direction estimation as well as fiber tracts reconstruction.

The rest of the paper is organized as follows. Section 2.2 provides background material for some common tensor models. The proposed methods for tensor direction estimation, smoothing of estimated directions, and fiber tracking are presented in, respectively, Sections 2.3, 2.4 and 2.5. Theoretical support for the direction smoothing method are presented in Section 2.6. The empirical performance of the overall methodology is illustrated with numerical experiments in Section 2.7 and with a real data set in Section 2.8. Section 2.9 provides some concluding remarks, while additional results and technical details are collected in an supplementary (Appendix A).

2.2 Tensor models

Suppose dMRI measurements are made on N voxels on a three dimensional grid representing a brain. For each voxel, we have measurements of diffusion weighted signals (complex numbers) along a fixed set (i.e., the same for all voxels) of unit-norm gradient vectors $\mathcal{U} = \{\mathbf{u}_i : i = 1, \ldots, m\}$.

By assuming Gaussian additive noise on both real and imaginary parts of the signal, the observed signal intensity can be modeled as

$$S(\mathbf{s}, \mathbf{u}) = \|\bar{S}(\mathbf{s}, \mathbf{u})\boldsymbol{\phi}(\mathbf{s}, \mathbf{u}) + \sigma\boldsymbol{\epsilon}(\mathbf{s}, \mathbf{u})\|,$$

where $\bar{S}(\mathbf{s}, \mathbf{u})$ is the intensity of the noiseless signal, $\boldsymbol{\phi}(\mathbf{s}, \mathbf{u})$ is a unit vector in $\mathbb{R}^2$ representing the phase of the signal, $\boldsymbol{\epsilon}(\mathbf{s}, \mathbf{u})$ is the noise random variable following

$\mathcal{N}_2(\mathbf{0}, \mathbf{I}_2)$ and $\sigma > 0$ denotes the noise level. The observed signal intensity then follows a Rician distribution (Gudbjartsson and Patz, 1995):

$$S(\mathbf{s}, \mathbf{u}) \sim \text{Rician}(\bar{S}(\mathbf{s}, \mathbf{u}), \sigma).$$

Moreover, we assume the noise $\boldsymbol{\epsilon}(\mathbf{s}, \mathbf{u})$'s are independent across different voxels and gradient directions. We write the set of measurements as $\{S(\mathbf{s}, \mathbf{u}) : \mathbf{u} \in \mathcal{U}\}$, where $\mathbf{s}$ is the three dimensional coordinate of the center of this voxel.

Assuming Gaussian diffusion, the noiseless signal intensity is given by (e.g., Mori, 2007)

$$\bar{S}(\mathbf{s}, \mathbf{u}) = S_0(\mathbf{s}) \exp\left\{-b\mathbf{u}^\intercal \mathbf{D}(\mathbf{s})\mathbf{u}\right\},$$

where $S_0(\mathbf{s})$ is the non-diffusion-weighted intensity, $b > 0$ is an experimental constant referred to as the b-value and $\mathbf{D}(\mathbf{s})$ is a 3×3 covariance matrix referred to as the diffusion tensor. This model is called the single tensor model and suits for the case of at most one dominant diffusion direction within a voxel. To indicate the degree of anisotropy of the diffusion, one commonly used measure is the fractional anisotropy (FA),

$$FA = \sqrt{\frac{(\lambda_1 - \lambda_2)^2 + (\lambda_2 - \lambda_3)^2 + (\lambda_3 - \lambda_1)^2}{2(\lambda_1^2 + \lambda_2^2 + \lambda_3^2)}}, \tag{2.1}$$

where λ_1, λ_2 and λ_3 are the eigenvalues of $\mathbf{D}$. FA value lies between zero and one and the larger it is, the more anisotropic the water diffusion is at the corresponding voxel.

Although the single tensor model is probably the most widely used tensor model in practice (implemented by most softwares for DTI), it is not suitable for crossing fiber regions. To deal with crossing fibers, this model has been extended to a multi-tensor model (e.g., Behrens *et al.*, 2007, 2003; Tabelow *et al.*, 2012; Tuch, 2002):

$$\bar{S}(\mathbf{s}, \mathbf{u}) = S_0(\mathbf{s}) \sum_{j=1}^{J(s)} p_j(\mathbf{s}) \exp\left\{-b\mathbf{u}^\intercal \mathbf{D}_j(\mathbf{s})\mathbf{u}\right\}, \tag{2.2}$$

where $\sum_{j=1}^{J(\mathbf{s})} p_j(\mathbf{s}) = 1$ and $p_j(\mathbf{s}) > 0$ for $j = 1, \ldots, J(\mathbf{s})$. Here $J(\mathbf{s})$ represents the number of fiber populations and $p_j(\mathbf{s})$'s denote weights of the corresponding fibers.

2.3 Voxel-wise estimation of diffusion directions

One important goal of DTI studies is to estimate principal diffusion directions, referred to as diffusion directions hereafter, at each voxel. They may be interpreted as tangent directions along fiber bundles at the corresponding voxel. The estimated diffusion directions are then used as an input for tractography algorithms to reconstruct fiber tracts. This section explores the diffusion direction estimation within a single voxel. For notational simplicity, dependence on voxel index $\mathbf{s}$ is temporarily dropped. Moreover, for ease of exposition, we assume that σ and $S_0(\mathbf{s})$ are known and delay the discussion of their estimation to Section 2.8.

Under the single tensor model, various methods for tensor estimation have been proposed including linear regression, nonlinear regression and ML estimation; e.g., see Carmichael *et al.* (2013) for a comprehensive review. Then diffusion directions are derived as principal eigenvectors of (estimated) diffusion tensors. However, for the multi-tensor models, severe computational issues have been observed and additional prior information and assumptions are imposed to tackle these issues. For instance, Behrens *et al.* (2007, 2003) use shrinkage priors and Tabelow *et al.* (2012) assume all tensors to be axially symmetric (i.e., the two minor eigenvalues are the same) and have the same set of eigenvalues. Scherrer and Warfield (2010) show that the multi-tensor model is indeed non-identifiable and they suggest to use multiple b-values in data acquisition to make the model identifiable. However, due to practical limitations, most of the current dMRI studies are obtained under a fixed b-value and so render their suggestion inapplicable. Below we show that the identifiability issue does not prevent one from estimating the diffusion directions and so neither strong assumptions nor special experimental settings are necessary if one is only interested in diffusion directions rather than the diffusion tensors themselves.

2.3.1 Identifiability of multi-tensor model

From Scherrer and Warfield (2010), model (2.2) can be re-written as

$$\bar{S}(\mathbf{u}) = S_0 \sum_{j=1}^{J} p_j a_j \exp\left\{-b\mathbf{u}^\intercal \left(\mathbf{D}_j + \frac{\log a_j}{b}\mathbf{I}_3\right)\mathbf{u}\right\},$$

where $a_j > 0$ for $j = 1, \ldots, J$ such that $p_j a_j > 0$, $\mathbf{D}_j + (\log a_j / b)\mathbf{I}_3$ is positive definite and $\sum_{j=1}^{J} p_j a_j = 1$. When $J = 2$, one can easily derive the explicit conditions for a_j to fulfill these criteria, and see that there are infinite sets of such a_j's. However, note that $\mathbf{D}_j + (\log a_j / b)\mathbf{I}_3$ shares the same set of eigenvectors with $\mathbf{D}_j$. Thus, one may still be able to estimate diffusion directions, which correspond to the major eigenvectors of the tensors. This motivates us to consider estimating diffusion directions directly instead of the tensors themselves.

Now we assume that $\mathbf{D}_j$'s are axially symmetric; that is, the two minor eigenvalues of $\mathbf{D}_j$ are equal. This is a common assumption (Basser *et al.*, 1994) for modeling dMRI data and it implies that diffusion is symmetric around the principal diffusion direction (here, the principal eigenvector) (Tournier *et al.*, 2007, 2004). By not differentiating the two minor eigenvectors, we obtain a clear meaning of diffusion direction. In addition, this reduces the number of unknown parameters by one and thus facilities estimation. In the following, we propose a new parametrization of the multi-tensor model which is identifiable and thus can be used for direction estimation.

Write $\mathcal{M}$ as the space of the unit principal eigenvector, i.e., the three dimensional unit sphere with equivalence relation $\mathbf{m} \sim -\mathbf{m}$. Let $\alpha_j \geq 0$, $\xi_j > 0$ and $\mathbf{m}_j \in \mathcal{M}$ be the difference between the larger and smaller eigenvalue, smaller eigenvalue and the standardized principal eigenvector of $\mathbf{D}_j$, respectively. Since $\mathbf{D}_j = \alpha_j \mathbf{m}_j \mathbf{m}_j^\intercal + \xi_j \mathbf{I}_3$, model (2.2) becomes

$$\begin{aligned}
\bar{S}(\mathbf{u}) &= S_0 \sum_{j=1}^{J} p_j \exp\left\{-b\mathbf{u}^\intercal \left(\alpha_j \mathbf{m}_j \mathbf{m}_j^\intercal + \xi_j \mathbf{I}_3\right) \mathbf{u}\right\} \\
&= S_0 \sum_{j=1}^{J} p_j \exp(-b\xi_j) \exp\left\{-b\alpha_j (\mathbf{u}^\intercal \mathbf{m}_j)^2\right\} \\
&= S_0 \sum_{j=1}^{J} \tau_j \exp\left\{-b\alpha_j (\mathbf{u}^\intercal \mathbf{m}_j)^2\right\}, \qquad (2.3)
\end{aligned}$$

where $\tau_j = p_j \exp(-b\xi_j) \in (0, 1)$. From the above, one can see that p_j and ξ_j are not simultaneously identifiable, so we cannot estimate the tensors. However, the new parametrization $\boldsymbol{\gamma} = (\boldsymbol{\gamma}_1^\intercal, \ldots, \boldsymbol{\gamma}_J^\intercal)^\intercal$ is identifiable, where $\boldsymbol{\gamma}_j = (\tau_j, \alpha_j, \mathbf{m}_j^\intercal)^\intercal$ for $j = 1, \ldots, J$, so that we can estimate the principal diffusion directions $\mathbf{m}_j$'s.

2.3.2 A fast and stable algorithm for ML estimation

To estimate the parameters in model (2.3), we start by investigating the standard ML estimation. Under the Rician noise assumption, the log-likelihood of $\boldsymbol{\gamma}$ is:

$$\begin{aligned} l(\boldsymbol{\gamma}) &= \sum_{\mathbf{u}\in\mathcal{U}} \log\left[\frac{S(\mathbf{u})}{\sigma^2}\exp\left\{-\frac{S^2(\mathbf{u})+\bar{S}^2(\mathbf{u})}{2\sigma^2}\right\} I_0\left\{\frac{S(\mathbf{u})\bar{S}(\mathbf{u})}{\sigma^2}\right\}\right] \\ &= \sum_{\mathbf{u}\in\mathcal{U}} \left[\log\left\{\frac{S(\mathbf{u})}{\sigma^2}\right\} - \frac{S^2(\mathbf{u})+\bar{S}^2(\mathbf{u})}{2\sigma^2} + \log I_0\left\{\frac{S(\mathbf{u})\bar{S}(\mathbf{u})}{\sigma^2}\right\}\right], \end{aligned} \tag{2.4}$$

where $I_0(x) = \int_0^\pi \exp(x\cos\phi)d\phi/\pi$ is the zeroth order modified Bessel function of the first kind (Abramowitz and Stegun, 1964). The ML estimate is obtained through maximizing the log-likelihood function (2.4). Although the new parametrization avoids the identifiability issue, the likelihood function usually has multiple local maxima, which makes the computation of ML estimate difficult and unstable. Next we discuss a strategy to tackle this issue.

In attempt to find the global maximizer, we develop an efficient algorithm through an approximation of model (2.3). This algorithm essentially performs a grid search, but it makes use of the geometry of the problem so it is quite fast. It includes three major steps: (i) lay down a grid for $(\alpha_j, \mathbf{m}_j^\intercal)$'s, (ii) evaluate the likelihood function on the grid, and (iii) return the grid point that maximizes the likelihood function. One can then use this returned grid point as a starting value in a gradient method for obtaining ML estimation of Model (2.3). Such a strategy results in better numerical stability and accuracy in finding ML estimates.

2.3.2.1 An approximation of model (2.3)

Let $\mathbf{c}_j = (\alpha_j, \mathbf{m}_j^\intercal)^\intercal$, $\mathbf{c} = (\mathbf{c}_1^\intercal, \ldots, \mathbf{c}_J^\intercal)^\intercal$ and $\mathcal{C}_j$ be the set of grid points for $\mathbf{c}_j$. For simplicity, we take the same set of grid points, $\mathcal{C}$, for all j. To lay down a grid for $\mathbf{m}_j$'s, we apply the sphere tessellation using Icosahedron, which is depicted in Figure 2.1. Here, we only pick unique vertices up to a sign for the formation of the grid. In our implementation, we utilize randomly rotated versions of the tessellation with two subdivisions, which results in a grid with 321 directions. If $\mathbf{c} \in \prod_{j=1}^J \mathcal{C}_j = \mathcal{C}^J$, model

(2.3) can be rewritten as

$$\bar{S}(\mathbf{u}) = \sum_{k=1}^{K} \tilde{\beta}_k x(\mathbf{u}, \tilde{\mathbf{m}}_k, \tilde{\alpha}_k), \tag{2.5}$$

where $K = |\mathcal{C}|$, $x(\mathbf{u}, \tilde{\mathbf{m}}_k, \tilde{\alpha}_k) = S_0 \exp\{-b\tilde{\alpha}_k(\mathbf{u}^\intercal \tilde{\mathbf{m}}_k)^2\}$, $(\tilde{\alpha}_k, \tilde{\mathbf{m}}_k) \in \mathcal{C}$ and $\tilde{\beta}_k \in [0, 1)$. One may notice that, in this reformulation, the non-zero $\tilde{\beta}_k$'s have an one-to-one correspondence with τ_j's in model (2.3). If $\mathbf{c} \notin \mathcal{C}^J$, i.e. the set of parameters is not a grid point, then equation (2.5) serves as an approximation to $\bar{S}(\mathbf{u})$ in model (2.3) as long as the grid is dense enough in the parameter space.

Furthermore, under the commonly used scales of b-values and tensors, $x(\mathbf{u}, \tilde{\mathbf{m}}_k, \tilde{\alpha}_k)$ and $x(\mathbf{u}, \tilde{\mathbf{m}}_{k'}, \tilde{\alpha}_{k'})$ are highly correlated if $\tilde{\mathbf{m}}_k = \tilde{\mathbf{m}}_{k'}$. Inspired by this observation, we reduce the grid size by setting $\tilde{\alpha}_k = \tilde{\alpha}$ for all k to a common value $\tilde{\alpha}$. From our experience, we set $\tilde{\alpha} = 2/b$. With all these approximations, we consider fitting the following model:

$$\bar{S}(u) = \sum_{k=1}^{K} \beta_k x_k(\mathbf{u}), \tag{2.6}$$

where $x_k(\mathbf{u}) = x(\mathbf{u}, \tilde{\mathbf{m}}_k, \tilde{\alpha})$ and $\beta_k \geq 0$. For our purpose, we want to identify nonzero β_k's because those $\tilde{\mathbf{m}}_k$'s associated with non-zero $\hat{\beta}_k$'s can be regarded as selected diffusion directions. Note that model (2.6) converts the expensive grid search to an estimation problem of a linear model (with respect to β_k's) with non-negative constraints. A fast algorithm for fitting this model with Rician noise assumption is given in Section S1 of the supplementary material (SM) (Appendix A). As it turns out, the non-negativity constraints often result in a sparse estimate of $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)^\intercal$; i.e., only a subset of directions is selected. In particular, if the estimate of the unconstrained problem (i.e., β_k's are allowed to be negative) is not located in the first quadrant of the parameter space, the corresponding constrained solution will be sparse.

Even though the solution is often sparse, the number of selected directions is usually larger than J, the true number of tensor components. This is partly due to colinearity of $x_k(\mathbf{u})$'s resulting from the use of a dense grid on the directions $\tilde{\mathbf{m}}_k$'s.

In the following, we propose to first divide the selected directions into I groups and then generate stable estimates of $\mathbf{m}_j$'s via gradient methods (Section 2.3.2.2). Fi-

nally, Bayesian information criterion (BIC) (Schwarz, 1978) is used to choose an appropriate I as the estimate for J (Section 2.3.3).

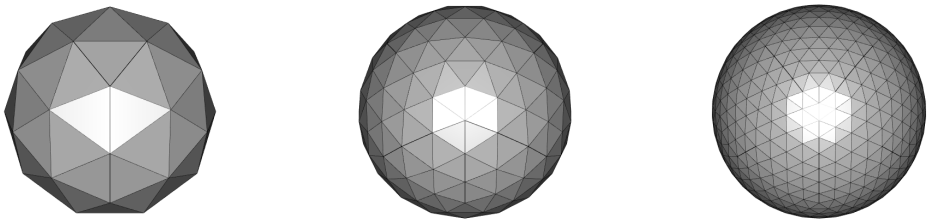


Figure 2.1. Sphere tessellations through triangulation using Icosahedron with level of subdivisions: 0 (Left), 1 (Middle) and 2 (Right).

2.3.2.2 Clustering of the selected directions

Write the above ML estimate of β_k as $\hat{\beta}_k$ for $k = 1, \ldots, K$. Suppose there are $L > 0$ nonzero $\hat{\beta}_k$'s, without loss of generality, $k = 1, \ldots, L$. Thus, $\tilde{\mathbf{m}}_1, \ldots, \tilde{\mathbf{m}}_L$ are the selected directions. Now, we develop a strategy to cluster the selected directions into I groups, for a set of $I \in \{1, \ldots, L\}$. To perform clustering, we require a metric measure on the space of directions $\mathcal{M}$. A natural metric is

$$d^*(\mathbf{u}, \mathbf{v}) = \arccos(|\mathbf{u}^\intercal \mathbf{v}|), \tag{2.7}$$

where $\mathbf{u}, \mathbf{v} \in \mathcal{M}$. Note that, $d^*(\mathbf{u}, \mathbf{v})$ is the acute angle between $\mathbf{u}$ and $\mathbf{v}$. With this distance metric, one can define dissimilarity matrix for a set of directions and make use of a generic clustering algorithm. Our choice is the Partition Around Medoids (PAM) (Kaufman and Rousseeuw, 1990) due to its simplicity. The detailed procedure is described in Algorithm S1 in the SM, where the input vectors are the selected directions. Due to the sparsity of $\hat{\beta}_j$'s and efficient algorithms of PAM, this clustering strategy is practically fast. Let $\breve{\mathbf{m}}_1, \ldots, \breve{\mathbf{m}}_I$ be the resulting group (Karcher) means. They are used as the starting value for gradient-based methods, such as L-BFGS-B algorithm (Byrd *et al.*, 1995), for obtaining $\hat{\gamma}(I)$, the ML estimate of γ under model (2.3) with I tensor components. More specifically, the starting value is set as $((1/I, \tilde{\alpha}, \breve{\mathbf{m}}_1^\intercal), \ldots, (1/I, \tilde{\alpha}, \breve{\mathbf{m}}_I^\intercal))^\intercal$.

2.3.3 Selection of the number of tensor components J

We use BIC to select the number of components J. Under model (2.3), the BIC for a model with I components is

$$\mathsf{BIC}(I) = -2l(\hat{\boldsymbol{\gamma}}(I)) + 4I\log(m), \tag{2.8}$$

where m is the number of gradient directions. Then J is chosen as

$$\hat{J} = \text{argmin}_{I\in\{1,\ldots,\tilde{I}\}}\mathsf{BIC}(I),$$

where $\tilde{I}$ is a pre-specified upper bound for the number of components. Based on our experience, $\tilde{I} = 4$ is reasonable choice. If $\tilde{I} > L$, we simply compare $\mathsf{BIC}(1), \ldots,$ $\mathsf{BIC}(L)$.

In practice, there are voxels with no major diffusion directions. Under single tensor model, the corresponding diffusion tensor is isotropic, i.e., all three eigenvalues are equal.

In the case of isotropic tensor, (2.3) reduces to $\bar{S}(\mathbf{u}) = \mathbf{S}_0\tau_1$. Thus there is only one parameter τ_1. We write the corresponding likelihood function as $\tilde{l}$ and denote the ML estimate of τ_1 by $\hat{\tau}_1$, which can be obtained by a generic gradient method. The corresponding BIC criterion is

$$\mathsf{BIC}(0) = -2\tilde{l}(\hat{\tau}_1) + \log(m),$$

where 0 represents no diffusion direction. Combined with the previous BIC formulation (2.8), one has a comprehensive model selection rule, which handles voxels with from zero to up to $\tilde{I}$ (here 4) fiber populations. In practice, we follow the convention and use FA (2.1) (see, e.g., Mori, 2007) as a first step screening; i.e., we first remove voxels with small FA values and then apply the BIC approach over those suspected anisotropic voxels.

We summarize our voxel-wise estimation procedure in Algorithm S2 in the SM. A simulation study is conducted and the corresponding results are presented in Section S2 of the SM. These numerical results suggest that our voxel-wise estimation procedure provides extremely stable and reliable results under various model settings.

2.4 Spatial smoothing of diffusion directions

Although model (2.3) provides a better modeling than single tensor model for crossing fiber regions, it also leads to an increase in the number of parameters and thus the variability of the estimates. To further improve estimation, we consider borrowing information from neighboring voxels and develop a novel smoothing technique for diffusion directions.

Tensor smoothing has been widely studied in the literature (Arsigny *et al.*, 2006; Carmichael *et al.*, 2013; Pennec *et al.*, 2006; Yuan *et al.*, 2012). However, as discussed earlier, tensors are not identifiable in the multi-tensor model without additional assumptions. Moreover, if the ultimate goal is the reconstruction of fiber tracts, a good estimate of diffusion directions suffices. This motivates the proposal of the new direction smoothing method below.

We shall assume that tangent directions of fiber bundles change smoothly. This leads to the spatial smoothness of diffusion directions that belong to the same fiber bundle. In many brain regions, it is reasonable to model the fiber tracts as smooth curves at the resolution of voxels in dMRI ($\sim$ 2mm).

However, there is one major challenge in diffusion direction smoothing. The smoothness assumption is only reasonable along the same fiber bundle. In regions with crossing fibers, the diffusion directions may belong to several different fibers which contribute to a mixture of populations of diffusion directions. To circumvent this issue, we propose to first cluster the diffusion directions within a neighborhood into separate homogeneous populations (Section 2.4.2) and then apply direction smoothing within each population (Section 2.4.1).

2.4.1 Smoothing along a single fiber

This subsection assumes that there is only one homogeneous population of diffusion directions, which corresponds to a single fiber bundle. Let $\{\{\hat{\mathbf{m}}_j(\mathbf{s}) : j = 1, \ldots, \hat{J}(\mathbf{s})\} : \forall \mathbf{s}\}$ be the estimated diffusion directions obtained from the above voxel-wise estimation procedure in Section 2.3. Further, write $T = \sum_{\mathbf{s}} \hat{J}(\mathbf{s})$ and, by re-indexing, $\{\hat{\mathbf{m}}_k : k = 1, \ldots, T\} = \{\{\hat{\mathbf{m}}(\mathbf{s}) : j = 1, \ldots, \hat{J}(\mathbf{s})\} : \forall \mathbf{s}\}$. Also write $\mathbf{s}_k$ as the correspond-

ing voxel location associated with $\hat{\mathbf{m}}_k$. Following the idea of kernel smoothing on Euclidean space (Fan and Gijbels, 1996), the smoothing estimate at voxel $\mathbf{s}_0$ is defined as a weighted Karcher mean of the neighboring direction vectors:

$$\arg\min_{\mathbf{v}\in\mathcal{M}} \sum_{i=1}^{T} w_i d^{*2}(\hat{\mathbf{m}}_i, \mathbf{v}), \tag{2.9}$$

where $w_i = K_{\mathbf{H}}(\mathbf{s}_i - \mathbf{s}_0)$'s are spatial weights and the metric d^* is defined in (2.7). These weights place more emphasis on spatially closer observations. Here $K_{\mathbf{H}}(\cdot) = |\mathbf{H}|^{-1/2}K(\mathbf{H}^{-1/2}\cdot)$ with $K(\cdot)$ as a three dimensional kernel function satisfying $\int K(\mathbf{s})d\mathbf{s} = 1$, and $\mathbf{H}$ is a 3×3 bandwidth matrix. In our numerical work, we choose K as the standard Gaussian density, and set $\mathbf{H} = h\mathbf{I}_3$, where h is chosen automatically using the cross-validation approach described in Section S3 of the SM.

2.4.2 Smoothing over multiple fibers

As discussed earlier, the spatial smoothness assumption does not hold in a voxel $\mathbf{s}_0$ with crossing fibers. To tackle this issue, we first apply clustering to estimated directions within a neighborhood of $\mathbf{s}_0$ in an attempt to separate the direction vectors corresponding to different fiber populations into different clusters. Then we apply the smoothing procedure in the previous subsection within each direction cluster. This subsection describes this procedure in details.

First we define neighboring voxels for $\mathbf{s}_0$. We begin with computing the spatial weights defined in Section 2.4.1. We then remove those voxels with weights smaller than a threshold. By filtering out these voxels, we obtain tighter and better separated clusters of directions. Moreover, such voxels have little effects on smoothing due to their small weights. The artificial data set displayed in Figure 2.2 provides an illustrative example. Every black dot in the left panel represents an estimated direction (from the center of the sphere). In the middle panel, the size of each dot is proportional to its spatial weight in equation (2.9). Lastly, the right panel shows all dots with spatial weights larger than a threshold. Notice that such a trimming operation leads to two obvious clusters of directions, which makes the subsequent task of clustering the estimated directions much easier.

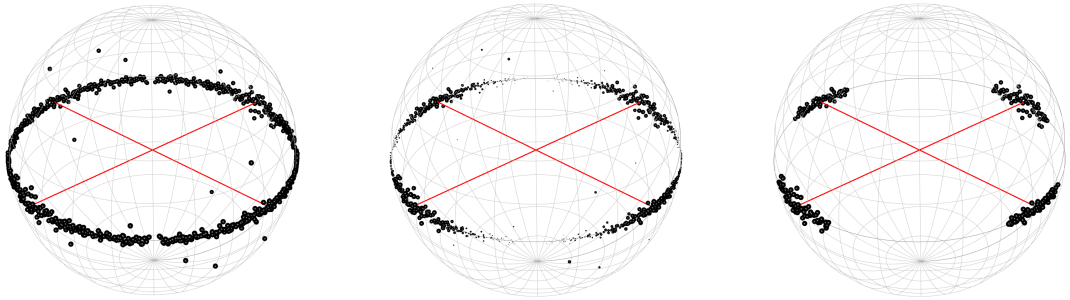


Figure 2.2. Finding all the neighboring voxels for separating crossing fiber directions. Left: all estimated directions. Middle: sizes of all estimated directions proportional to weights. Right: estimated directions with weights larger than a threshold. Red lines represents underlying true directions.

Next we utilize the same clustering strategy developed in Section 2.3.3 to choose the number of clusters adaptively by the average silhouette (Rousseeuw, 1987); see Algorithm S3 of the SM. The silhouette of a datum i measures the strength of its membership to its cluster, as compared to the neighboring cluster. Here, the neighboring cluster is the one, apart from cluster of datum i, that has the smallest average dissimilarity with datum i. The corresponding silhouette is defined as $(b_i - a_i)/(\max\{a_i, b_i\})$, where a_i and b_i represent the average dissimilarities of datum i with all other data in the same cluster and that with the neighboring cluster respectively. The average silhouette of all data gives a measure of how good the clustering is. Thus we select the number of clusters via maximizing the average silhouette.

The detailed smoothing procedure is given in Algorithm 1.

2.5 Fiber tracking

For dMRI, fiber tractography can be divided into deterministic and probabilistic methods. Deterministic methods (e.g. Mori *et al.*, 1999; Mori and van Zijl, 2002; Weinstein *et al.*, 1999) track fiber bundles by utilizing the principal eigenvectors of tensors. Probabilistic methods (e.g. Friman *et al.*, 2006; Koch *et al.*, 2002; Parker and Alexander, 2003) use the probability density of diffusion orientations. Deterministic methods, including the popular Fiber Assignment by Continuous Tracking (FACT) (Mori *et al.*,

Algorithm 1 Algorithm for direction smoothing

Input: Target voxel $\mathbf{s}^*$, voxel-wise estimate $\{(\mathbf{s}_k, \hat{\mathbf{m}}_k), k = 1, \ldots, T\}$, estimated number of fibers $\{\hat{J}(\mathbf{s}) : \mathbf{s} \in \mathcal{S}\}$, kernel function K, bandwidth matrix $\mathbf{H}$, threshold c, maximum number of cluster (Algorithm S3 of the SM) K, angular threshold (Algorithm S3 of the SM) ξ

Output: Updated number of directions and updated directions at $\mathbf{s}^*$

Description: To perform smoothing for diffusion directions at $\mathbf{s}^*$

1: **for** $k = 1$ to T **do** Compute spatial weight: $w_k \leftarrow K_{\mathbf{H}}(\mathbf{s}_k - \mathbf{s}^*)$

2: **for** $k = 1$ to T **do** Standardize spatial weights: $w_k \leftarrow w_k / \sum_{j=1}^{T} w_j$

3: Sort w_k's in decreasing order such that $w_{l_1} \geq \cdots \geq w_{l_T}$

4: Identify neighborhood for clustering (Section 2.4.2):
Compute $L \leftarrow \min_{M \in \{1,\ldots,T\}} \mathbb{1}\{\sum_{m=M+1}^{T} w_{l_m} \leq c\}$ (The summation $\sum_{m=T+1}^{T} w_{l_m}$ is defined as 0.)

5: Clustering via Algorithm S3 (SM): $(\{\mathbf{u}_1, \ldots, \mathbf{u}_C\}, C) \leftarrow$CLUSTDIRN($\{\hat{\mathbf{m}}_{l_1}, \ldots, \hat{\mathbf{m}}_{l_L}\}$, K, ξ)

6: **if** $C \geq \hat{J}(\mathbf{s}^*)$ **then**

7: Match the smoothed directions, $\{u_1, \ldots, u_C\}$, to the voxel-wise estimates at $\mathbf{s}^*$, $\{\hat{\mathbf{m}}_1(s^*), \ldots, \hat{\mathbf{m}}_{\hat{J}(s^*)}(s^*)\}$:

$$\left(\hat{k}_1, \ldots, \hat{k}_{\hat{J}(\mathbf{s}^*)}\right) \leftarrow \underset{\{k_1 \ldots, k_{\hat{J}}(\mathbf{s}^*) \in \{1,\ldots,C\} : k_i \neq k_j\}}{\arg\min} \sum_{j=1}^{\hat{J}(\mathbf{s}^*)} d^*(\hat{\mathbf{m}}_j(\mathbf{s}^*), \mathbf{u}_{k_j})$$

8: **for** $j = 1$ to $\hat{J}(\mathbf{s}^*)$ **do** $\hat{\mathbf{m}}_j(\mathbf{s}^*) \leftarrow \mathbf{u}_{\hat{k}_j}$

9: **else**

10: Match the voxelwise estimates at $\mathbf{s}^*$ to the smoothed directions: :

$$\left(\hat{k}_1, \ldots, \hat{k}_C\right) \leftarrow \underset{\{k_1 \ldots, k_C \in \{1,\ldots,\hat{J}(\mathbf{s}^*)\} : k_i \neq k_j\}}{\arg\min} \sum_{j=1}^{C} d^*(\hat{\mathbf{m}}_{k_j}(\mathbf{s}^*), \mathbf{u}_j)$$

11: **for** $j = 1$ to C **do** $\hat{\mathbf{m}}_{\hat{k}_j}(\mathbf{s}^*) \leftarrow \mathbf{u}_j$

12: $\hat{J}(\mathbf{s}^*) \leftarrow C$ and remove non-updated $\hat{\mathbf{m}}_j(\mathbf{s}^*)$'s

1999) and Tensorlines (Weinstein *et al.*, 1999) algorithms, typically require a diffusion tensor field, where there is a single diffusion tensor (either isotropic or anisotropic) associated with each voxel, as an input. In below, we propose a deterministic tracking algorithm which takes diffusion directions (associated with the location information) as input. This algorithm allows for multiple or no principal diffusion directions at a voxel. One advantage of the proposed algorithm is that it makes use of the directional information from individual fibers at voxel level.

To construct our procedure, we adopt similar tracking ideas from FACT, as depicted in Figure 2.3 (Left). Tracking starts at the center of a voxel (Voxel 1 in Figure 2.3) and continues in the direction of the estimated diffusion direction. When it enters the next voxel (Voxel 2 in Figure 2.3), the track changes its direction to align with the new diffusion direction and so on. The above tracking rule may produce many short and fragmented fiber tracts due to either a wrongfully identified isotropic voxel or spurious directions which go nowhere. In addition, it does not tell us which direction to follow in case there are multiple directions in a voxel, which happens in crossing fiber regions. To address these issues, we modify the above procedure in the following.

Given a current diffusion direction (we refer to the corresponding voxel as the current voxel), the voxel that it points to (we refer to this voxel as the destination voxel) may have (i) at least one direction; (ii) no direction (i.e., isotropic). In case (i), we will first identify the direction with the smallest angular difference with the current direction. If its separation angle is smaller than a pre-specified threshold (e.g., $\pi/6$), we enter the destination voxel and tracking will go on along this direction. See Figure 2.3 (Middle). On the other hand, if the separation angle is greater than the threshold, or case (ii) happens, we deem that the destination voxel does not have a viable direction. In this case, tracking will go along the current direction if it finds a viable direction within a pre-specified number of voxels. The number of voxels that are allowed to be skipped is set to be 1 in our numerical illustrations. See Figure 2.3 (Right). On the other hand, the tracking stops at the current voxel if no viable

directions within a pre-specified number of voxels can be found. The detailed tracking algorithm is described in Algorithm S4 in the SM.

As for the choice of starting voxels (known as seeds), there are two common strategies. One can choose seeds based on tracts of interest and starts the tracking from a region of interest (ROI). This approach is based on knowledge on ROI and may not give a full picture of the tracts of interest if there are diverging branches. The other approach is called brute-force approach, where tracking starts from every voxel. It usually leads to a more comprehensive picture of tracts at a higher computational cost.

The proposed algorithm can be coupled with either strategy. In the brute-force approach, we apply Algorithm S4 of the SM for every pair of $(\mathbf{s}_k, \hat{\mathbf{v}}_k)$ twice, i.e., $(\mathbf{s}_k, \hat{\mathbf{v}}_k)$ and $(\mathbf{s}_k, -\hat{\mathbf{v}}_k)$. Due to the continuity of fiber, one would not expect a fiber going to and from nowhere, and only exists within a single voxel. Therefore, if that happens, we remove the corresponding fiber.

The simplicity of the proposed algorithm makes various extensions possible. For instance, we may use weighted average of neighboring directions to produce smoother tracts, similar to Mori and van Zijl (2002).

Combining with the aforementioned smoothing procedure, we call the resulting technique **Di**ffusion **di**ection **S**moothing and **T**racking (DiST).

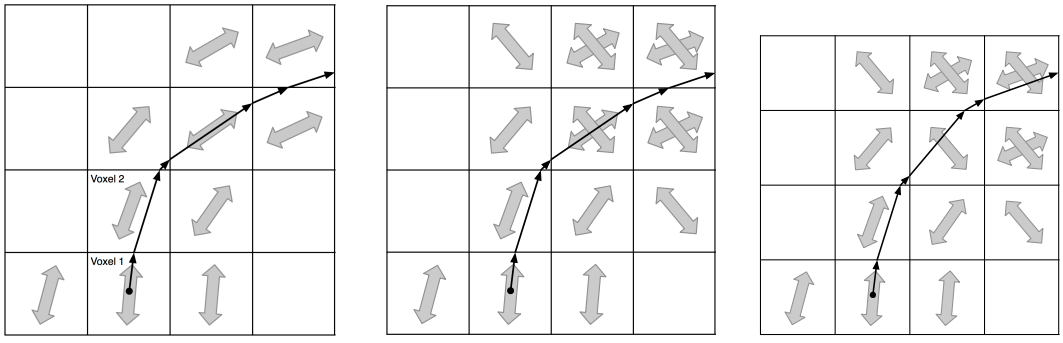


Figure 2.3. Left: Demonstration of the proposed algorithm in single fiber region. Middle: Demonstration of the proposed algorithm in crossing fiber region. Right: Demonstration of the proposed algorithm in case of absence of viable directions.

2.6 Theoretical results

This section derives some asymptotic properties of the proposed direction smoothing estimator. Note that, since the space of direction vectors has a non-Euclidean geometry and so the theoretical framework is different from that of classical smoothing estimators. Without loss of generality, suppose we observe $\mathbf{v}_1, \ldots, \mathbf{v}_n \in \mathcal{M}$ at spatial locations $\mathbf{s}_1, \ldots, \mathbf{s}_n$ respectively. Let $\mathcal{V}$ be the three dimensional unit sphere. Then $\mathcal{M}$ is the quotient space of $\mathcal{V}$ with equivalence relation $\mathbf{v} \sim -\mathbf{v}$ for any $\mathbf{v} \in \mathcal{V}$. This space is also identified with the so-called real projective space $\mathbb{R}P^2$.

In the following, we derive our theoretical results under random design where $\mathbf{s}_i$'s are independently and identically sampled from a distribution with density f_S, but our results also apply to regular voxels. Given a spatial location $\mathbf{s}_0$, our target is to estimate $\mathbf{v}_0$, namely the diffusion direction at $\mathbf{s}_0$, in the sense that it minimizes $\mathbb{E}\left\{d^{*2}(\mathbf{V}, \mathbf{v}) | \mathbf{S} = \mathbf{s}_0\right\}$, where $d^*(\mathbf{u}, \mathbf{v}) = \arccos(|\mathbf{u}^\intercal \mathbf{v}|)$. For simplicity, we assume $\mathbf{s}_i \in \mathbb{R}$ and write it as s_i thereafter. Thus, our estimator (2.9) at s_0 can be written as

$$\hat{\mathbf{v}}(s_0) = \arg\min_{\mathbf{v} \in \mathcal{M}} \sum_{i=1}^{n} K_h(s_i - s_0) d^{*2}(\mathbf{v}_i, \mathbf{v}),$$

where $K_h(\cdot) = K(\cdot/h)/h$. Here, with slight notation abuse, $K(\cdot)$ represents a one dimensional kernel function throughout the theoretical developments.

2.6.1 Working coordinate system

For each $\mathbf{p} \in \mathcal{V}$, one can endow a tangent space $T_{\mathbf{p}}\mathcal{V} = \{\mathbf{v} \in \mathbb{R}^3 : \mathbf{v}^\intercal \mathbf{p} = 0\}$ with the metric tensor $g_{\mathbf{p}} : T_{\mathbf{p}}\mathcal{V} \times T_{\mathbf{p}}\mathcal{V} \to \mathbb{R}$ defined as $g_{\mathbf{p}}(\mathbf{u}_1, \mathbf{u}_2) = \mathbf{u}_1^\intercal \mathbf{u}_2$. Note that the tangent space is identified with $\mathbb{R}^2$. The geodesics are great circles and the geodesic distance is $\arccos(\mathbf{p}_1^\intercal \mathbf{p}_2)$, for any $\mathbf{p}_1, \mathbf{p}_2 \in \mathcal{V}$. The corresponding exponential map at $\mathbf{p} \in \mathcal{V}$, $\mathrm{Exp}_{\mathbf{p}} : T_{\mathbf{p}}\mathcal{V} \to \mathcal{V}$, is given by

$$\mathrm{Exp}_{\mathbf{p}}(\mathbf{0}) = \mathbf{p} \quad \text{and} \quad \mathrm{Exp}_{\mathbf{p}}(\mathbf{u}) = \cos(\|\mathbf{u}\|)\mathbf{p} + \frac{\sin(\|\mathbf{u}\|)}{\|\mathbf{u}\|}\mathbf{u} \quad \text{when} \quad \mathbf{u} \neq \mathbf{0},$$

while the corresponding logarithm map at $\mathbf{p} \in \mathcal{V}$, $\mathrm{Log}_{\mathbf{p}} : \mathcal{V} \backslash \{-\mathbf{p}\} \to T_{\mathbf{p}}\mathcal{V}$, is given by

$$\mathrm{Log}_{\mathbf{p}}(\mathbf{p}) = \mathbf{0} \quad \text{and} \quad \mathrm{Log}_{\mathbf{p}}(\mathbf{v}) = \frac{\arccos(\mathbf{v}^\intercal \mathbf{p})}{\sqrt{1 - (\mathbf{v}^\intercal \mathbf{p})^2}}[\mathbf{v} - (\mathbf{v}^\intercal \mathbf{p})\mathbf{p}] \quad \text{when} \quad \mathbf{v} \neq \mathbf{p}.$$

One can use the exponential map and the logarithm map to define a coordinate system for the $\mathcal{V}\backslash\{-\mathbf{v}_0\}$ in the following way. Given $\mathbf{v} \in \mathcal{V}$, we define the logarithmic coordinate as

$$\omega_1 = \mathbf{e}_1^\intercal \mathrm{Log}_{\mathbf{v}_0}(\mathbf{v}) \quad \text{and} \quad \omega_2 = \mathbf{e}_2^\intercal \mathrm{Log}_{\mathbf{v}_0}(\mathbf{v}),$$

where $\mathbf{e}_1, \mathbf{e}_2 \in T_{\mathbf{v}_0}\mathcal{V}$ and $\{\mathbf{e}_1, \mathbf{e}_2\}$ forms an orthonormal basis for $T_{\mathbf{v}_0}\mathcal{V}$. Write $\phi(\mathbf{v}) = (\omega_1, \omega_2)^\intercal$. In addition, we define

$$\rho_{\mathbf{v}_0}(\mathbf{v}) = \begin{cases} \mathrm{sign}(\mathbf{v}_0^\intercal \mathbf{v})\mathbf{v} & \mathbf{v}_0^\intercal \mathbf{v} \neq 0 \\ \mathbf{v} & \mathbf{v}_0^\intercal \mathbf{v} = 0 \end{cases},$$

and

$$d(\boldsymbol{\omega}, \boldsymbol{\theta}) = d^*(\tilde{\phi}^{-1}(\boldsymbol{\omega}), \tilde{\phi}^{-1}(\boldsymbol{\theta})), \qquad \boldsymbol{\omega}, \boldsymbol{\theta} \in \mathbb{R}^2,$$

where $\tilde{\phi} = \phi \circ \rho_{\mathbf{v}_0}$. Here, we define $\rho_{\mathbf{v}_0}^{-1}$ as an identity map.

2.6.2 Asymptotic results

Now, write $\boldsymbol{\theta}_i = \tilde{\phi}(\mathbf{v}_i)$ for $i = 1, \ldots, n$, and $\psi(\boldsymbol{\omega}, \boldsymbol{\theta}) = d^2(\boldsymbol{\omega}, \boldsymbol{\theta})$. We have $\boldsymbol{\theta}_0 = \tilde{\phi}(\mathbf{v}_0) = \mathbf{0}$. Also, let $\psi_1(\boldsymbol{\omega}, \boldsymbol{\theta})$ and $\psi_2(\boldsymbol{\omega}, \boldsymbol{\theta})$ be the first and second order derivative of ψ with respect to $\boldsymbol{\theta}$ respectively. Let $\mathbf{m}(s) = (m_1(s), m_2(s))^\intercal = \mathbb{E}(\boldsymbol{\theta}_1 | S_1 = s)$ and $\boldsymbol{\Sigma}(s) = [\Sigma_{jk}(s)]_{1 \leq j,k \leq 2} = \mathrm{Var}(\boldsymbol{\theta}_1 | S_1 = s)$. Also, denote $\boldsymbol{\Psi}(s) = [\Psi_{jk}(s)]_{1 \leq j,k \leq 2} = \mathbb{E}[\psi_2(\boldsymbol{\theta}_1, \boldsymbol{\theta}_0) | S_1 = s]$. Write $\mathcal{B}_\delta(\boldsymbol{\theta}_0) = \{\boldsymbol{\theta} \in \mathbb{R}^2 : \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\| < \delta\}$, for $\delta > 0$. Throughout our discussion, we use the L_2-norm for matrix. We need the following assumptions to proceed.

Assumption 2.1. *There exists $\epsilon > 0$ such that* $\mathrm{supp}(\mathbf{V}_1 | S_1 = s) \subseteq \{\mathbf{v} \in \mathbb{R}^3 : d^*(\mathbf{v}, \mathbf{v}_0) \leq \pi/2 - \epsilon\}$, *in a neighborhood of s_0.*

Assumption 2.2. *$h \to 0$ and $nh \to \infty$.*

Assumption 2.3. *$K(\cdot)$ is bounded, compactly supported kernel satisfying (i) $\int K(x)dx = 1$ and (ii) $\int xK(x)dx = 0$.*

Assumption 2.4. *The density of S, $f_S(\cdot)$, is twice continuously differentiable in a neighborhood of s_0 and $f_S(s_0) > 0$.*

Assumption 2.5. *$m_j(\cdot)$ is twice continuously differentiable in a neighborhood of s_0, for $j = 1, 2$.*

Assumption 2.6. *$\Sigma_{jk}(\cdot)$ is continuous in a neighborhood of s_0, for $j, k = 1, 2$.*

Assumption 2.7. *$\Psi_{jk}(\cdot)$ is continuous in a neighborhood of s_0, for $j, k = 1, 2$.*

Assumption 2.8. *$E\{[\psi_2(\boldsymbol{\theta}_1, \boldsymbol{\theta}_0)]^2_{j,k} | S_1 = s\} \leq C_{jk}$ for all s, for $j, k = 1, 2$.*

Assumption 2.9. *Let $\gamma(\delta, s) = \mathbb{E}[\sup_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \|\psi_2(\boldsymbol{\theta}_1, \tilde{\boldsymbol{\theta}}) - \psi_2(\boldsymbol{\theta}_1, \boldsymbol{\theta}_0)\| | S_1 = s]$. There exists a neighborhood of s_0, $\mathcal{W}(s_0)$, such that*

$$\tilde{\gamma}(\delta) = \sup_{s \in \mathcal{W}(s_0)} \gamma(\delta, s) = o(1),$$

as $\delta \to 0$.

Assumption 2.10. *$\boldsymbol{\Psi}(s_0)$ is positive definite.*

Assumption 1 is a technical assumption for avoiding the unnecessary complication arising from the representation of geodesic distance as a function of the working coordinate system. As a result of Assumption 1, one can use a representation of $\pm\mathbf{v}$, which aligns with $\mathbf{v}_0$, and reduces the geodesic distance of $\mathcal{M}$ to the geodesic distance of $\mathcal{V}$. This assumption is usually satisfied by our procedure, as a results of thresholding and clustering. Assumptions 2-10 are standard conditions for consistency and distributional limits for smoothing estimators.

Theorem 2.1. *Suppose Assumptions 1-10 hold. Let $M_n(\boldsymbol{\theta}) = \sum_{i=1}^n hK_h(S_i - s_0) d^2(\boldsymbol{\theta}_i, \boldsymbol{\theta})$.*

(a) There exists a sequence of solutions, $\hat{\boldsymbol{\theta}}_n(s_0)$, to $M_n^{(1)}(\boldsymbol{\theta}) = 0$, such that $\hat{\boldsymbol{\theta}}_n(s_0)$ converges in probability to $\boldsymbol{\theta}_0$.

(b) And $\hat{\boldsymbol{\theta}}_n$ is asymptotically normal:

$$\sqrt{nh}\left\{(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) - h^2\boldsymbol{\eta}\right\} \implies \mathcal{N}_2(\mathbf{0}, \boldsymbol{\Omega}),$$

where

$$\boldsymbol{\eta} = 2\int x^2 K(x)dx\boldsymbol{\Psi}^{-1}(s_0)\left\{\frac{f_S^{(1)}(s_0)}{f_S(s_0)}m^{(1)}(s_0) + \frac{1}{2}m^{(2)}(s_0)\right\}$$

and

$$\boldsymbol{\Omega} = 4\int K^2(x)dx\boldsymbol{\Psi}^{-1}(s_0)\boldsymbol{\Sigma}(s_0).$$

The proof of the Theorem 2.1 can be found in Section S5 of the SM.

2.7 Simulation results

This section presents simulation results of the proposed DiST procedure. For simulation results of the voxel-wise estimation procedure proposed in Section 2.3, see Section S2 of the SM.

We simulate 200 diffusion tensor data sets from the tensor field given in Figure 2.4 (Top). The tensors all have the principal eigenvalues being 4×10^{-3} and FA (2.1) being 0.9. The b-value is set to be 1000 across all voxels. This mimics the b-value and diffusivity (reflected by the numerical scale of the tensor) in a real dMRI study.

At each voxel there is either one tensor or there are two tensors. For crossing fiber regions, p_1 and p_2 are set to 0.7 and 0.3 respectively, and the separation angles between the two tensors range from 66.3 to 86.6 degree. In crossing fiber regions of Figure 2.4 (Top), the more transparent the tensor is, the less weight it takes.

In addition, $S_0(\mathbf{s})$'s have the same value which is set to 1000. Two choices of the noise standard deviation σ are used, namely 50 and 100, which corresponds to signal-to-noise ratio (S_0/σ) of 20 and 10, respectively. The case that $\mathsf{SNR} = 20$ is typical for

dMRI studies while that SNR $= 10$ corresponds to a high noise setting. The set of gradient directions $\mathcal{U}$ is obtained from the sphere tessellation with 3 subdivision using octahedron and $|\mathcal{U}| = 33$, which is in a typical range for dMRI studies nowadays. With these gradient directions, the observed signal intensities $\mathbf{S}(\mathbf{s})$'s are simulated according to the multi-tensor model (2.2) with the Rician noise. A total of four different procedures are compared:

- raw: voxel-wise estimation without any smoothing;
- DiST-cv: DiST using ordinary cross-validation score for choosing h;
- DiST-tcv: DiST using 5% trimmed cross-validation score for choosing h;
- DiST-mcv: DiST using median cross-validation score for choosing h.

See Section S3 of the SM for definitions of the various cross-validation variants.

Table 2.1 shows numerical summaries of the simulation results. In addition to the proportion of correctly estimated number of diffusion directions, we also report the mean MSE (MMSE) and the mean root MSE (MRMSE), defined as follows. Conditional on the correct estimation of J, the squared error of $\mathbf{m}$ is defined as

$$\min_{\{k_1,\ldots,k_J\in\{1,\ldots,J\}:k_i\neq k_j\}} \sum_{j=1}^{J} d^{*2}(\mathbf{m}_j, \hat{\mathbf{u}}_{k_j}), \tag{2.10}$$

where $\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_J$ are the estimated diffusion directions. Here, the MSE is the mean of squared errors (2.10) over voxels with $\hat{J} = J$ in one simulated data set and root MSE (RMSE) is the square root of MSE. Then MMSE and MRMSE are defined, respectively, as the means of MSEs and RMSEs over the 200 simulated data sets.

The voxel-wise estimation works reasonably well in estimating both the number of diffusion directions J and the diffusion directions. Even for the low SNR setting, the correctness of estimation of J is around 75% and the angular error is no more than 11 degree. For the single tensor region ($J = 1$), smoothing improves upon estimation of both J and diffusion directions. For regions with two tensors ($J = 2$), smoothing only

improves direction estimation. Among the three smoothing procedures, DiST-mcv works the best.

Table 2.2 shows the five-number summary of the maximum angular error with $\hat{J} = J = 2$ across the 200 simulated data sets. Again smoothing procedures have smaller errors than the raw procedure and DiST-mcv is the best among all methods. For DiST-mcv, the mean and median of angular errors are around 2.5 degree and 1 degree for SNR $= 10$ and SNR $= 20$, respectively. Such magnitude of errors has little impact on tracking.

We then apply the proposed tracking algorithm in Section 2.5 (Algorithm S4, SM) to the estimated diffusion directions based on the above procedures. The tracking results of a simulation with SNR = 10 are shown in Figures 2.4 (Bottom) and 2.5. As can be seen in Figure 2.5, the lines produced by DiST are much more aligned when compared to the tracking result based on voxel-wise estimation without smoothing (raw).

Table 2.1. Diffusion direction estimation results. **Correct-select**: proportion of $\hat{J} = J$. **MMSE**: mean of MSEs (Each MSE is computed over voxels with $\hat{J} = J$ in one simulated data set.), in squared degree, of the estimated diffusion direction, with the corresponding standard error stated in brackets. **MRMSE**: mean of RMSEs (Each RMSE is computed over voxels with $\hat{J} = J$ in one simulated data set.), in degree, of the estimated diffusion direction, with the corresponding standard error stated in brackets.

SNR	J		raw	DiST-cv	DiST-tcv	DiST-mcv
10	1	Correct-select	97.12%	99.09%	99.15%	99.45%
		MMSE	9.84 (3.84e-02)	4.95 (2.94e-01)	2.70 (1.09e-01)	3.06 (1.40e-01)
		MRMSE	3.14 (6.12e-03)	2.09 (5.46e-02)	1.60 (2.60e-02)	1.69 (3.13e-02)
	2	Correct-select	75.18%	74.38%	75.37%	75.44%
		MMSE	114 (2.42)	50.9 (3.45)	40.0 (3.11)	9.81 (1.40)
		MRMSE	10.6 (1.07e-01)	6.05 (2.68e-01)	5.26 (2.49e-01)	2.49 (1.35e-01)
20	1	Correct-select	98.59%	99.46%	99.69%	99.75%
		MMSE	2.30 (8.50e-03)	1.25 (1.23e-01)	7.97e-01 (3.02e-02)	1.15 (5.47e-02)
		MRMSE	1.52 (2.80e-03)	1.02 (3.28e-02)	8.79e-01 (1.10e-02)	1.03 (2.04e-02)
	2	Correct-select	99.38%	99.94%	99.99%	99.99%
		MMSE	19.8 (2.12e-01)	6.43 (5.18e-01)	2.00 (2.84e-01)	1.48 (2.13e-01)
		MRMSE	4.43 (2.34e-02)	2.13 (9.75e-02)	1.13 (6.02e-02)	9.93e-01 (4.98e-02)

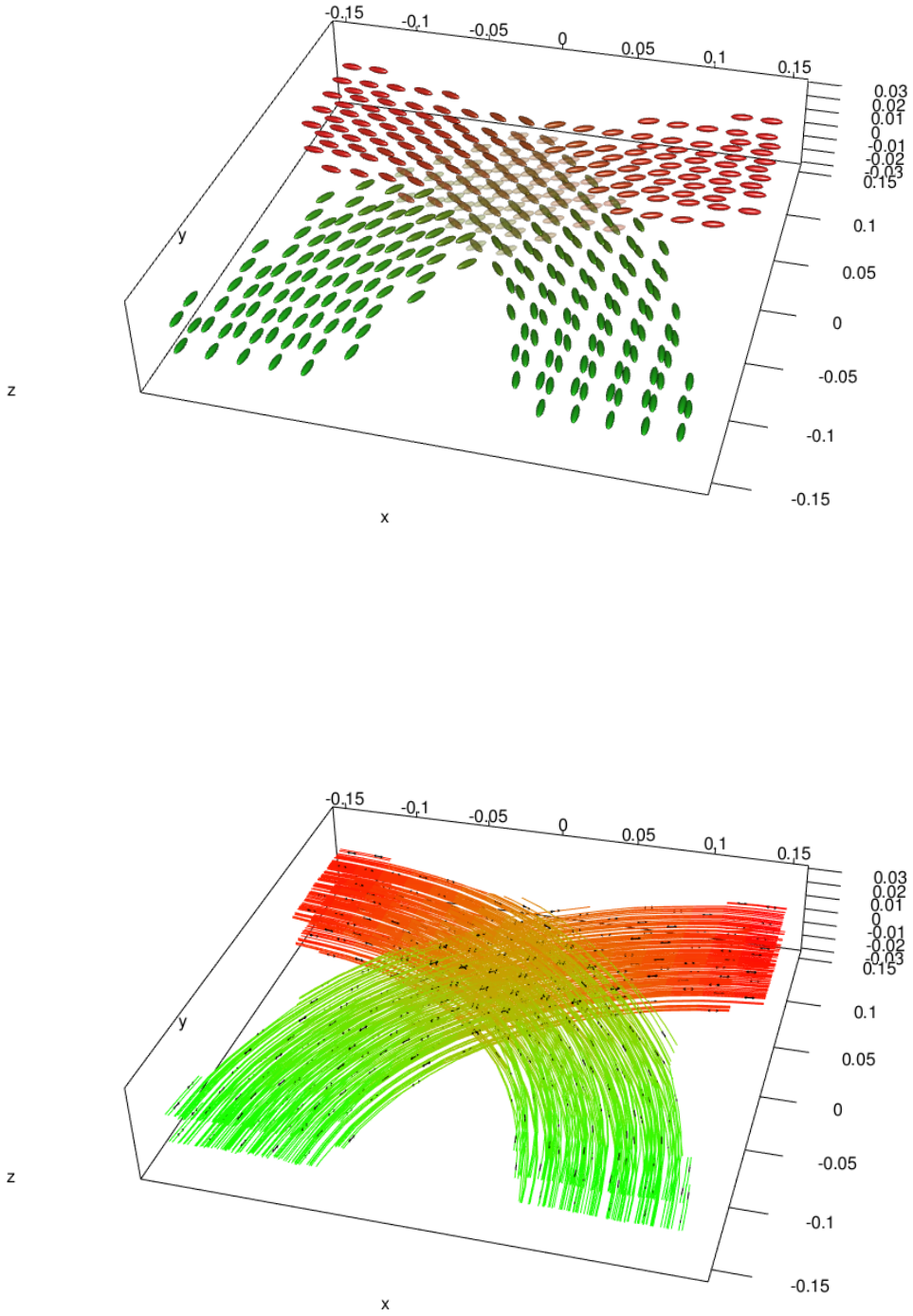


Figure 2.4. Top: The true tensor field used in the simulation study (Section 2.7). Bottom: Illustration of fiber tracking using DiST-mcv.

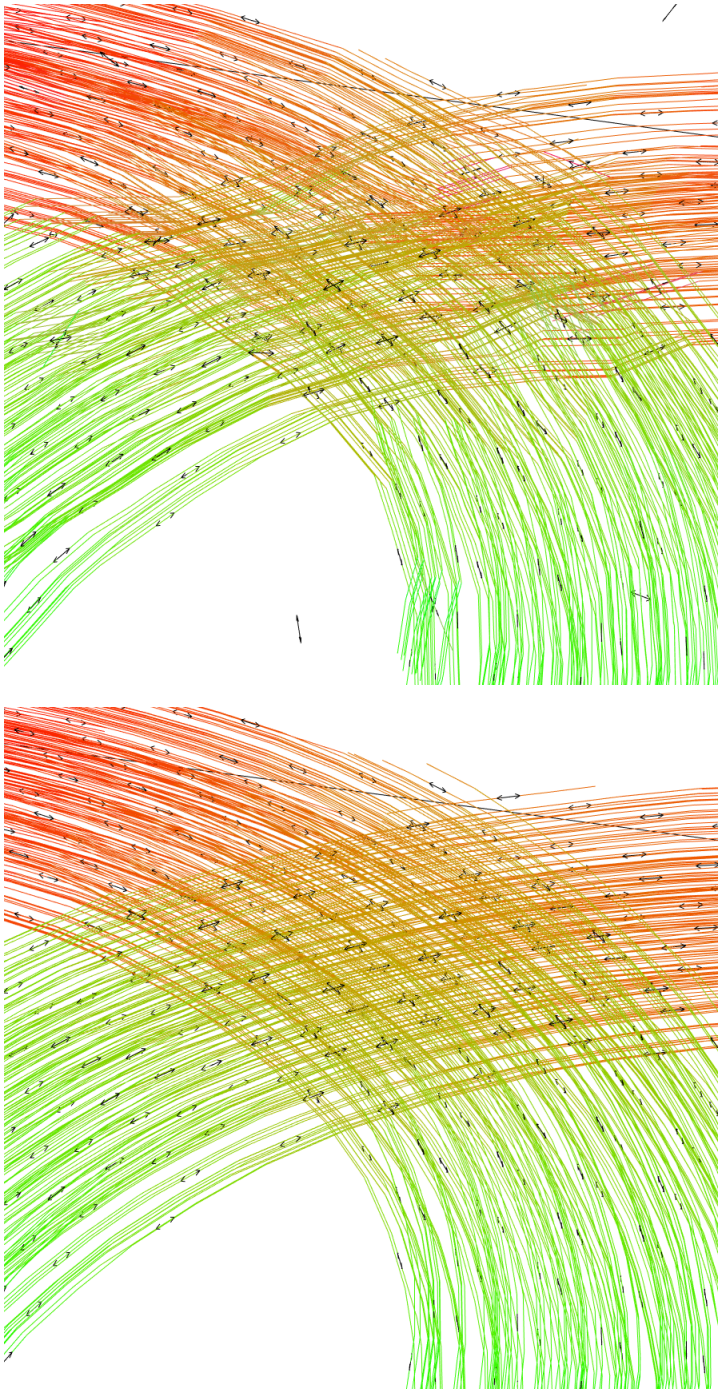


Figure 2.5. Illustration of fiber tracking over the crossing fiber region by raw (top) and DiST-mcv (bottom) respectively.

Table 2.2. Summary statistics of the maximum absolute error across the voxels with $\hat{J} = J = 2$.

SNR	Method	Minimum	1st Quantile	Median	Mean	3rd Quantile	Maximum
10	raw	0.530	6.63	9.86	11.8	14.6	89.3
	DiST-cv	0.132	2.32	4.99	6.97	9.59	89.3
	DiST-tcv	0.0933	2.08	4.01	6.00	8.07	89.3
	DiST-mcv	0.135	1.35	2.11	2.91	3.35	65.1
20	raw	0.350	3.20	4.67	5.20	6.65	29.5
	DiST-cv	0.0803	0.931	1.73	2.48	3.28	26.1
	DiST-tcv	0.0494	0.613	0.965	1.33	1.53	15.7
	DiST-mcv	0.0473	0.531	0.841	1.16	1.40	15.9

2.8 Real data application

In this section, we apply the proposed methodology to a real dMRI data set, which was obtained from the Alzheimer's Disease Neuroimaging Initiative (ADNI) database (www.loni.ucla.edu/ADNI). The primary goal of ADNI has been to test whether serial MRI, positron emission tomography (PET), other biological markers, and clinical and neuropsychological assessment can be combined to measure the progression of mild cognitive impairment (MCI) and onset of Alzheimer's disease (AD). In the following, we use an eddy-current-corrected ADNI data set of a normal subject for illustration of our technique.

This data set contains 41 distinct gradient directions with b-value set as $1000 s/mm^2$. In addition, there are 5 $b0$ images (corresponding to $b = 0$), forming in total 46 measurements for each of the $256 \times 256 \times 59$ voxels. To implement our technique, we require estimates of $S_0(\mathbf{s})$'s and σ. We first estimate $S_0(\mathbf{s})$ and $\sigma(\mathbf{s})$ for each voxel by ML estimation based on the 5 $b0$ images. Then we fix σ as the median of estimated $\sigma(\mathbf{s})$'s for voxel-wise estimation of the diffusion directions. Since the original $256 \times 256 \times 59$ voxels contain volume outside the brain, we only take median over a human-chosen set of $81 \times 81 \times 20$ voxels. The estimated σ is 56.9.

In this analysis, we focus on a subset of voxels ($15 \times 15 \times 5$), which contains the intersection of corpus callosum (CC) and corona radiata (CR). This region is known to contain significant fiber crossing (Wiegell *et al.*, 2000). Figure 2.6 shows the fiber

orientation color map (derived from the single tensor model). The aforementioned region is indicated by a white rectangular box. Within this region, $S_0(\mathbf{s})$'s have mean 1860.1 and standard deviation 522.7.

We then apply voxel-wise estimation to individual voxels followed by the DiST-mcv procedure. Distributions of the estimated number of diffusion directions are summarized in Table 2.3. For comparison purposes, we also fit the single tensor model with the commonly used regression estimator (e.g., Mori, 2007).

The tracking results are produced by applying the proposed tracking algorithm to the estimated diffusion directions from DiST-mcv and those from the single tensor model estimation. Figure 2.7 shows the corresponding tracking results. For visualization purposes, we also present the longest 300 tracts in Figure 2.8. From anatomy, the CC has a mediolateral direction while the CR has a superoinferior orientation. They are clearly shown in both tracking results. In these figures, reconstructed fiber tracts are colored by a RGB color model with red for left-right, green for anteroposterior, and blue for superior-inferior. Thus, one can easily locate the CC and the CR as the red fiber bundle and the blue fiber bundle respectively. Tracking result based on DiST-mcv shows clear crossing between mediolateral fiber and the superoinferior fiber (In the figure, the crossing of red and blue fiber tracts). From neuroanatomic atlases and previous studies, Wiegell *et al.* (2000) conclude that there are several fiber populations with crossing structure in this conjunction region of CC and CR, which matches with the tracking based on DiST-mcv. However, the single tensor model estimation can only reconstruct one major diffusion direction in each voxel and thus the corresponding tracking result does not show crossing structure. Instead, the CC (red fiber bundle) is blocked by the CR (blue fiber bundle) and this leads to either termination of the CC fiber tracts or significant merging of the CC and the CR fiber tracts instead of the known crossing structure. To give further illustration, Figure 2.9 shows the locations of the CC, the CR and the region of crossing fibers (Cross). One can see that tracking based on DiST-mcv reproduces the crossing fiber structures between the CC and the CR, while the result based on single tensor model tends to connect the

CC and the CR fibers.

Moreover, the green fiber on top of the CC represents the cingulum bundle. Both fiber tracking based DiST and single fiber model produce clear and sensible reconstruction of cingulum bundle. All these features match with neuroanatomic atlases and provide a good demonstration of our proposed method.

Table 2.3. Number of voxels with different estimated number of diffusion directions.

	Number of diffusion directions				
	0	1	2	3	4
Voxel-wise estimation	37	476	589	23	0
Smoothing	37	476	593	19	0

2.9 Discussion

Using tensor estimation to resolve cross-fiber can be problematic, due to the non-identifiability issue in multi-tensor model. In this paper, we take a different route by focusing on the estimation of diffusion directions rather than the non-identifiable diffusion tensors. We develop the corresponding direction smoothing procedure and fiber tracking strategy, together called DiST, along this route. Our technique gives promising empirical results in both simulation study and real data analysis.

The procedure we presented works well even with moderate number of gradient directions (a few tens), as long as the number of distinct crossing fibers within a voxel is not large. With HARDI data, which can have up to a couple of hundreds gradient directions, rather than modeling the direction distribution within a tensor framework, we can estimate the fiber orientation distribution nonparametrically (Descoteaux *et al.*, 2007; Tuch, 2004). In that case, we can potentially extend the fiber tracking procedure presented here by adopting a probabilistic approach in which the directions for moving from one voxel to another are sampled from the fiber orientation distribution. Such a probabilistic fiber tracking has the additional advantage of giving a measure of uncertainty of the fiber tracts extracted from the data. This is a topic of future research.

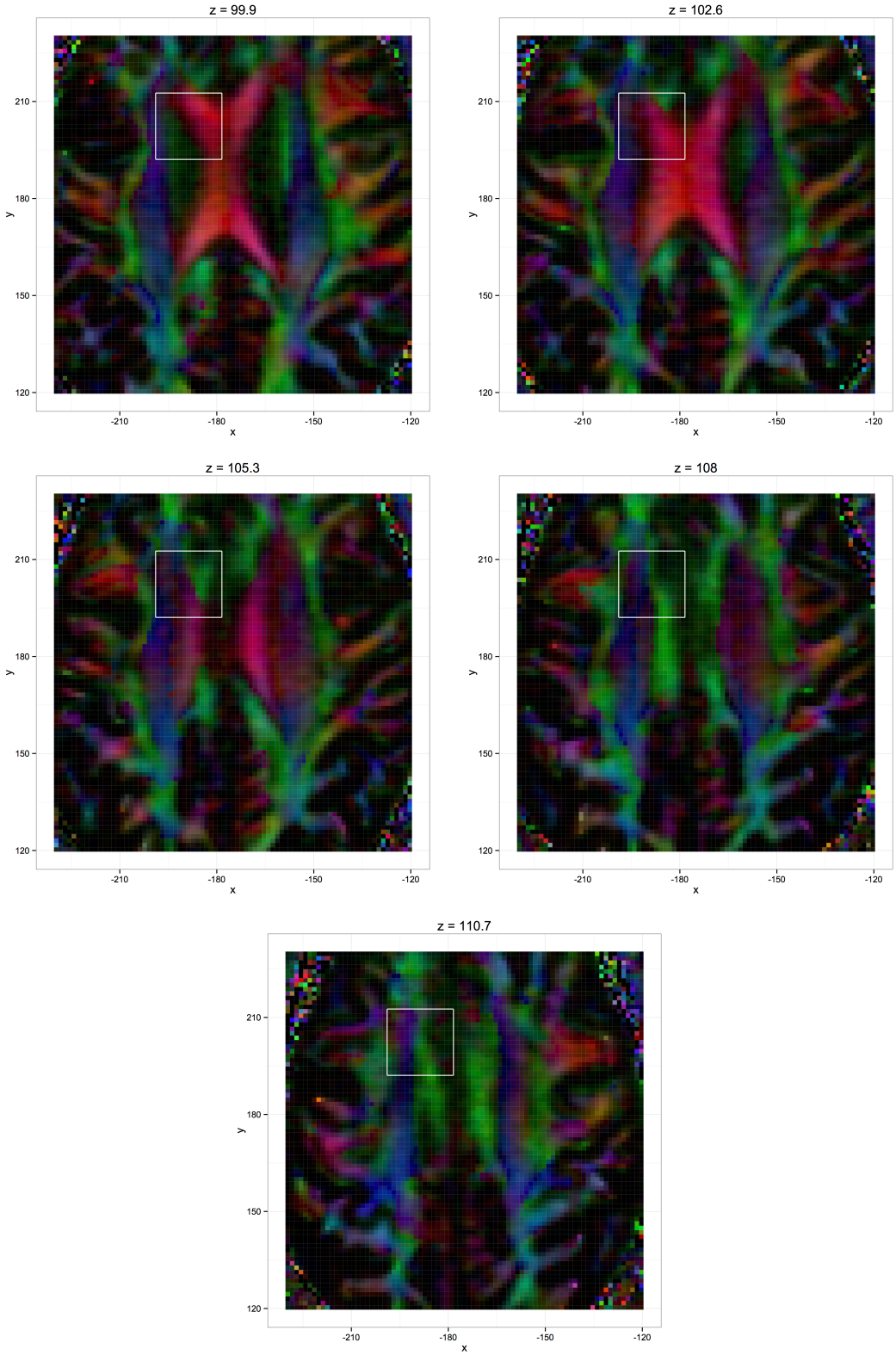


Figure 2.6. The fiber orientation color map (based on the single tensor model). The focused region is indicated by white rectangular boxes.

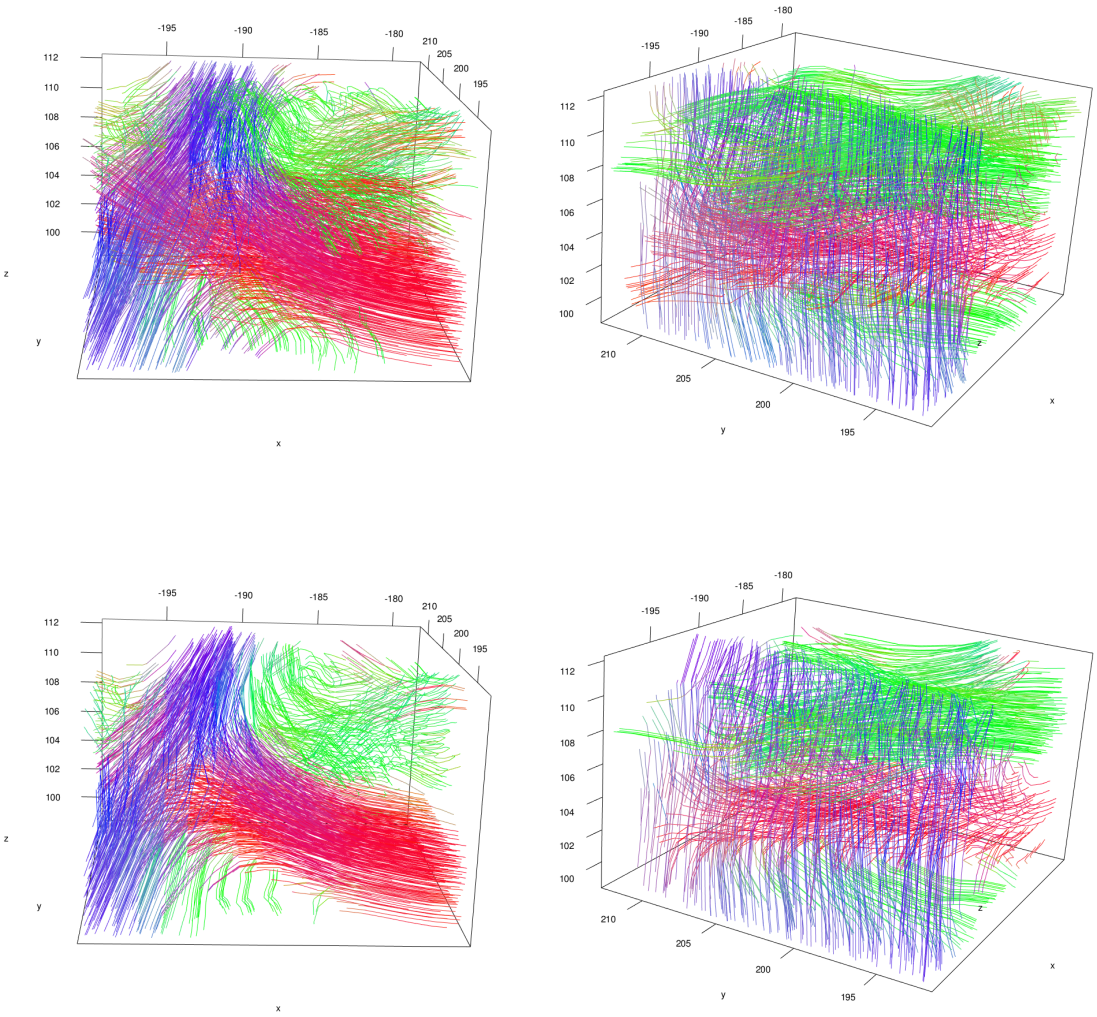


Figure 2.7. Top: Tracking using DiST-mcv. Bottom: Tracking using single tensor model.

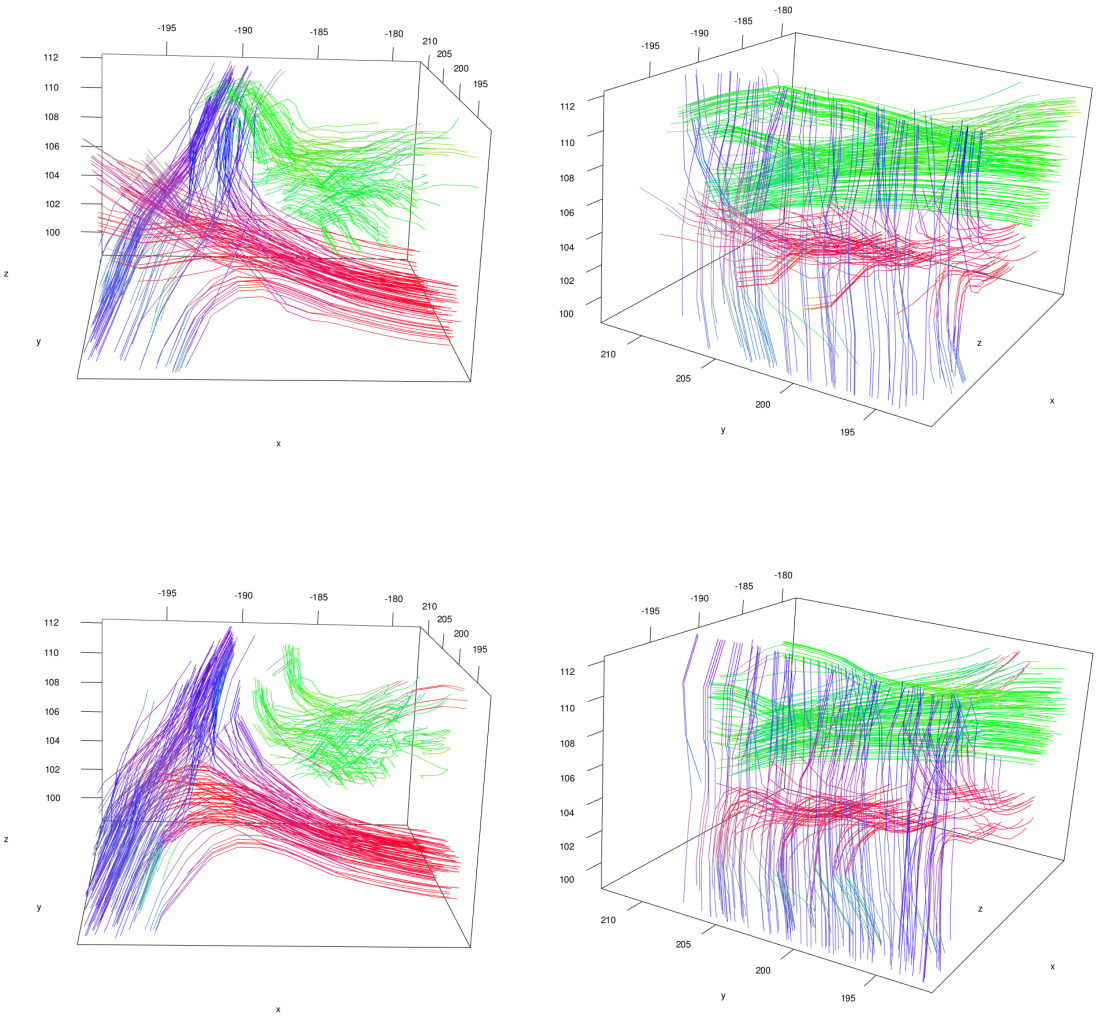


Figure 2.8. Top: The longest 300 tracks using DiST-mcv (The left and right figures corrspond to different view angles). Bottom: Similarly for single tensor model.

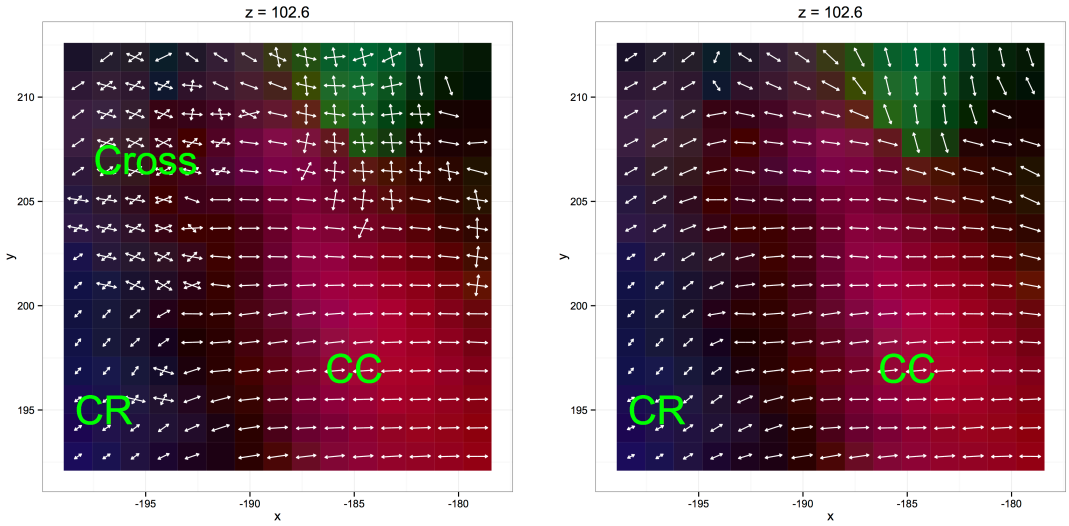


Figure 2.9. The projection of fiber directions to the xy-plane at $z = 102.6$ for illustration of crossing fibers. The plot also shows the location of corpus callosum (CC), corona radiata (CR) and crossing region (Cross). The fiber orientation color map is overlayed as the background. Left: for DiST-mcv. Right: for single tensor model.

Chapter 3

Global Optimization of High Dimensional Expensive Black-box Systems with Uncertainty Quantification

Abstract

Numerical optimization is usually conducted through iterative evaluations of a target function. In many situations, such as computer experiments, the target function is complicated and the corresponding evaluations are costly. Emulators (i.e., statistical response surfaces) can be used to overcome these difficulties. One common strategy to handle such situations is to adaptively sample the input space for a new set of target function evaluations with which to update the emulator. This is accomplished by involving another optimization of a quantity (e.g., expected improvement) as a function of the next set of locations for evaluation. However, evaluation and optimization of this quantity can be challenging when the input dimension of the function becomes large. In this work, we propose a novel technique to overcome the aforementioned difficulties via explorations of sparsity. This technique can generate multiple sampling locations (batch sampling) so that evaluations of the target function can be done in a parallel manner. As another promising feature, our technique provides uncertainty quantification

of the optimal solution, which can be used to safeguard scenarios of multiple global optima or occurrences of competitive local optima.

This is a joint work with Curtis B. Storlie[1] and Thomas C. M. Lee[2].

3.1 Introduction

In many situations such as computer experiments, the underlying target functions (e.g. computer models) are complicated and costly to evaluate. Such target functions are usually called black-box functions. Despite the vast study of global optimization, optimizations of these functions are still challenging. Such optimizations are very different from the traditional global optimization framework, under which function evaluations are relatively cheap. It is very common that a single evaluation of expensive functions, such as those in computer experiments, takes a few days with reasonably fast machines. Moreover, the black-box functions are complicated and so, very often, no simple model can be laid down for them. In addition, the evaluations are often imperfect and subject to noise contamination, such as in a manufacturing process. In the examples when the target function is a computer model, the noise is relatively small, but they usually are not negligible because numerical solving error of the computational code results in a little bit of jitter around the true value.

In many problems related to expensive black-box functions, statistical nonparametric surface estimation plays an important role. These surface estimates, called emulators, provides a flexible modeling for black-box functions. A common optimization strategy is to adaptively sample the input space for a new set of target function evaluations based on the current surface estimate. This is accomplished by involving another optimization of a quantity such as expected improvement (Jones *et al.*, 1998). This falls into the framework of sequential sampling. The purpose of sequential sampling is to make the best possible use of the model evaluations for the purpose at hand. For example, if we are interested in obtaining the minimum of the target function then sampling at places where we know the function is likely to be

[1]Los Alamos National Laboratory
[2]Department of Statistics, University of California at Davis

high (given what we know so far) is of little use to us. Instead, we should observe at values of inputs where the function could be small. Thus some have used a sequential sampling scheme where they throw down an initial sample of size n_0, estimate the function given the n_0 points, then use some criterion, like expected improvement in the minimum, to determine a location to evaluate next. That means, one has to optimize the criterion with respect to the values of inputs. This nested optimization can be a curse in traditional regime where function evaluations are cheap. However, if function evaluations are expensive, the gain in utilizing the information of existing evaluations usually outweighs the additional computational power invested in obtaining a better set of next sampling locations.

The aforementioned strategy has been widely used (Ginsbourger *et al.*, 2010; Huang *et al.*, 2006a,b; Jones *et al.*, 1998). However, when the dimension of the input space becomes large, there are three major complications with this strategy:

1. Decent estimation of target function becomes challenging, so that calculating the criterion for where to evaluate next produces highly variable results.

2. Searching the input space for the point with the most favorable criterion value becomes very challenging, and suboptimal searches must be performed (with varying levels of how suboptimal they are the bigger dimension is).

3. Just calculating the criterion may require a high dimensional integral which makes the issue in 2 even worse.

In a lot of problems with high dimensionality, many input dimensions do not have strong effect on the target function value. So the idea is to use a surface estimation that has variable selection built in to select the useful input dimensions. The procedure we propose at a qualitative level is the following:

1. Lay down a Latin Hypercube Sample (LHS) and estimate the target function with a variable selection procedure.

2. Use some criterion (e.g., expected information increase) to determine where to place the next value(s) of inputs to evaluate the target function. This is a reduced

dimensional search because many of the dimensions of the inputs have no effect on the criterion due to the variable selection. The other dimensions are still sampled with an LHS sample or other sampling scheme. Thus, the other zeroed out dimensions *will* have a chance to come back.

3. Also the criterion is easier to compute for each candidate point because the expected information gain is only a function of the non-zero input variables.

In addition, the proposed procedure can generate multiple sampling locations at a time. This is called batch designs (e.g., Loeppky *et al.*, 2010). Batch design is very useful in situations like computer experiments, where the evaluations of the target function can be obtained in a parallel manner. With variable selection, the surface estimates are usually of lower dimensions and this makes the uncertainty quantification of the optimal solution computationally feasible. This can be used to safeguard scenarios of multiple global optima or occurrences of competitive local optima.

3.2 Emulator with variable selection

In this section, we focus on the surface estimation of a target function f, which is assumed to be smooth. As f is complicated and difficult to be modeled easily, the nonparametric surface estimation techniques have been widely used as surrogates or emulators of f. Our choice of emulator is ACOSSO (Storlie *et al.*, 2011), which is an extension of smoothing spline ANOVA (Gu, 2013; Wahba, 1990). Similarly as LASSO and adaptive LASSO (Tibshirani, 1996; Zou, 2006), ACOSSO has an embedded variable selection power, which can automatically remove 'weak' functional components. To lay down the mathematical framework, suppose

$$y(\boldsymbol{x}) = f(\boldsymbol{x}) + \varepsilon$$

where $f \in \mathcal{F}$ (a reproducing kernel Hilbert space) is the interested surface (e.g., a manufacturing process or a computer model) and ε is the observational error. Here the observational error follows $\mathcal{N}(0, \sigma^2)$. For different observations, we assume f stays the same, but the corresponding observational errors are independently and

identically distributed. Without loss of generality, we assume $\boldsymbol{x} = (x_1, \ldots, x_p)^\mathsf{T} \in \mathcal{X} = [0,1]^p$. And we take $\mathcal{F}$ as p-dimensional tensor product of some reproducing kernel Hilbert spaces on $[0,1]$, $\{\mathcal{F}_j\}_{j=1}^p$. One common choice is to take all $\mathcal{F}_j$'s as the second order Sobolev spaces, $\mathcal{S}^2 = \{g : g, g' \text{ are absolutely continuous and } g'' \in \mathcal{L}^2[0,1]\}$, with squared norm

$$\|f\|^2 = \left(\int_0^1 f(x)dx\right)^2 + \left(\int_0^1 f'(x)dx\right)^2 + \int_0^1 \left(f''(x)\right)^2 dx. \tag{3.1}$$

Write the reproducing kernel of $\mathcal{F}_j$ as K_j. Using the smoothing spline ANOVA decomposition (see, e.g., Gu, 2013; Wahba, 1990), one can express

$$\mathcal{F} = \bigotimes_{j=1}^p \mathcal{F}_j = \{1\} \oplus \left\{\bigoplus_{j=1}^p \bar{\mathcal{F}}_j\right\} \oplus \left\{\bigoplus_{j<k} (\bar{\mathcal{F}}_j \otimes \bar{\mathcal{F}}_k)\right\} \oplus \cdots$$

where $\{1\}$ represents the space of constant functions and $\mathcal{F}_j = \{1\} \oplus \bar{\mathcal{F}}_j$. With truncation, we can rewrite

$$\mathcal{F} = \{1\} \oplus \left\{\bigoplus_{j=1}^q \mathcal{F}_j\right\}.$$

In general, if we observe samples

$$y_i(\boldsymbol{x}_i) = f(\boldsymbol{x}_i) + \varepsilon_i, \qquad i = 1, \ldots, n,$$

ACOSSO selects $f \in \mathcal{F}$ that minimizes

$$\frac{1}{n}\sum_{i=1}^n \{y_i(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\}^2 + \lambda \sum_{j=1}^q w_j \|P^j f\|_{\mathcal{F}}, \tag{3.2}$$

where $P^j f$ is the orthogonal projection of f onto the $\mathcal{F}_j$ for $j = 1, \ldots, q$. Here the sum of norms gives rise to the sparsity of the solution (Storlie *et al.*, 2011).

Under the sequential framework, our strategy is to apply ACOSSO for summarizing the information from previous samples to help look for better sites for the next sample. Roughly speaking, the ACOSSO summarizes two pieces of information. First, it achieves variable selection. If the variable selection is correct, the unselected variables are essentially classified as irrelevant variables and so special sampling design

for these variables should not improve the situation. Thus, with variable selection, we can confine our focus to the construction of adaptive design of the low dimensional subspace of $\mathcal{X}$. Second, ACOSSO provides the adaptive kernels which are useful for deriving the design criterion. We will explain more about this in the next section.

3.3 Equivalent formulation and empirical Bayesian interpretation of ACOSSO

In this section, we study the empirical Bayesian interpretation of ACOSSO, which we can use for the construction of the criterion for sequential sampling. First, consider the problem of finding $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_q)^\intercal$ and $f \in \mathcal{F}$ to minimize

$$\frac{1}{n}\sum_{i=1}^{n}\{y_i(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\}^2 + \lambda_0 \sum_{j=1}^{q}\theta_j^{-1} w_j^2 \|P^j f\|_{\mathcal{F}}^2 + \lambda_1 \sum_{j=1}^{q}\theta_j, \qquad \text{subject to } \theta_j \geq 0 \ \forall j. \tag{3.3}$$

From Storlie *et al.* (2011), it is shown that, under $\lambda_1 = \lambda^2/(4\lambda_0)$, (3.2) and (3.3) shares the same minimizer $\hat{f}$ (with $\hat{\theta}_j = \lambda^{1/2}\lambda_1^{-1/2} w_j \|P^j \hat{f}\|_{\mathcal{F}}$, $j = 1, \ldots, q$, for (3.3)). For fixed $\boldsymbol{\theta}$, minimizing (3.3) is equivalent to minimizing

$$\frac{1}{n}\sum_{i=1}^{n}\{y_i(\boldsymbol{x}_i) - f(\boldsymbol{x}_i)\}^2 + \lambda_0 \sum_{j=1}^{q}\theta_j^{-1} w_j^2 \|P^j f\|_{\mathcal{F}}^2.$$

This corresponds to a classical smoothing spline estimation and the solution has the Bayesian interpretation as a posterior mean of $f(\boldsymbol{x}) = \mu + \sum_{j=1}^{q}(P^j f)(\boldsymbol{x})$ where $P^j f$ has a mean zero Gaussian process prior on $\mathcal{X}$ with covariance function $G_j(\boldsymbol{s}, \boldsymbol{t}) = \sigma^2 \theta_j w_j^{-2} K_j(\boldsymbol{s}, \boldsymbol{t})/n\lambda_0$, for $\boldsymbol{s}, \boldsymbol{t} \in \mathcal{X}$. Here K_j is the corresponding reproducing kernel with respect to $\mathcal{F}_j$. In ACOSSO, we estimate $\boldsymbol{\theta}$ from the data and the resulting estimate of f can be viewed as a smoothing spline estimate under $\boldsymbol{\theta} = \hat{\boldsymbol{\theta}}$. This idea is basically an empirical Bayesian approach.

The variable selection mechanism of ACOSSO lies in setting some θ_j's to zero. Under the Bayesian interpretation, the corresponding prior covariance functions G_j's are set to zero. Thus one can understand why the posterior mean of f has zero projected component on the corresponding $\mathcal{F}_j$.

This above interesting observation facilitates a corresponding Gaussian process model for f. Suppose now $P^j f$ has a mean zero Gaussian process prior on $\mathcal{X}$ with covariance function G_j, for $j = 1, \ldots, q$ and write $G = \sum_{j=1}^{q} G_j$. The distribution of observation $(y_1(\boldsymbol{x}_1), \ldots, y_n(\boldsymbol{x}_n))^\intercal$ is given by

$$\mathcal{N}\left\{\mu \mathbf{1}_n, \left(G(\boldsymbol{x}_i, \boldsymbol{x}_j)\right)_{i,j=1}^{n} + \sigma^2 \boldsymbol{I}_n\right\}.$$

And, suppose, now, we are given a set of new observations $y'_1(\boldsymbol{x}'_1), \ldots, y'_m(\boldsymbol{x}'_m)$. The joint distribution of $(y_1(\boldsymbol{x}_1), \ldots, y_n(\boldsymbol{x}_n), y'_1(\boldsymbol{x}'_1), \ldots, y'_m(\boldsymbol{x}'_m))^\intercal$ is given by

$$\mathcal{N}\left\{\mu \mathbf{1}_{n+1}, \begin{pmatrix} \left(G(\boldsymbol{x}_i, \boldsymbol{x}_j)\right)_{i,j=1}^{n} + \sigma^2 \boldsymbol{I}_n & \left(G(\boldsymbol{x}_i, \boldsymbol{x}'_j)\right)_{i,j=1}^{n,m} \\ \left(G(\boldsymbol{x}'_i, \boldsymbol{x}_j)\right)_{i,j=1}^{m,n} & \left(G(\boldsymbol{x}'_i, \boldsymbol{x}'_j)\right)_{i,j=1}^{m} + \sigma^2 \boldsymbol{I}_m \end{pmatrix}\right\}.$$

Write the covariance matrix as

$$\boldsymbol{R} = \begin{pmatrix} \boldsymbol{R}_{11} & \boldsymbol{R}_{12} \\ \boldsymbol{R}_{12}^\intercal & \boldsymbol{R}_{22} \end{pmatrix}$$

according to the above matrix partition. The posterior covariance matrix of $(y'_1(\boldsymbol{x}'_1), \ldots, y'_m(\boldsymbol{x}'_m))^\intercal$ can be written as $\boldsymbol{R}_{23} - \boldsymbol{R}_{12}^\intercal \boldsymbol{R}_{11}^{-1} \boldsymbol{R}_{12}$.

3.4 Kernel Proposal

Note that the Sobolev's norm (3.1) induces a kernel $K(s, t)$. In Figure 3.1, the plot of K is shown. For instance, one may see that when s is set as around 0.8, $K(0.8, t)$ is an increasing function with respect to t. Thus, $K(0.8, 1)$ is larger than $K(0.8, 0.8)$. Basically, that means the prior covariance between the $f_j(0.8)$ and $f_j(1)$ is larger than the variance of $f_j(0.8)$. This effect carries to the search of good design and result in more emphasis on the boundary, 0 and 1.

To solve this problem, the Gaussian kernel is used. In this case, we have to select the variance parameter of the Gaussian kernel. Note that the variance parameter is related to the smoothness of each dimension. For computational simplicity, the variance parameters for all dimensions are set to be the same and is chosen by Bayesian information criterion (Schwarz, 1978). Some properties of the Gaussian kernel as the

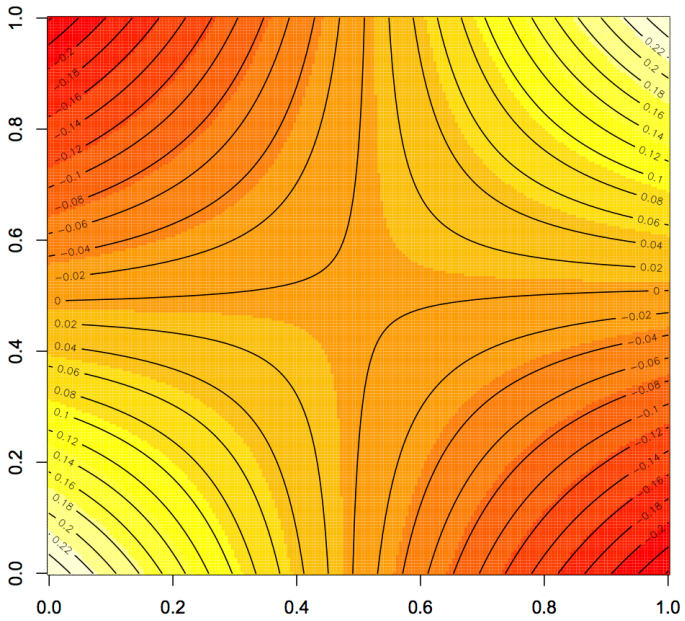


Figure 3.1. The demonstration of the induced kernel K from (3.1).

reproducing kernel are given by Steinwart *et al.* (2006) and Minh (2010). One important property is the reproducing kernel Hilbert space generated by this kernel does not include nonzero constant functions.

3.5 Sequential sampling for minimizing f

In this section, we aim to design a sequential sampling scheme for estimating the minimizer of f. For kriging models, expected improvement criterion is developed by Jones *et al.* (1998) to formulate the sequential sampling scheme for global optimization. Huang *et al.* (2006b) extend the expected improvement criterion to global optimization from noisy observations and call it augmented expected improvement criterion. To cope with batch sampling, Ginsbourger *et al.* (2010) explore a batch version of expected improvement criterion. But due to the additional computational burden, they propose using the constant liar algorithm which utilizes the original expected improvement criterion. Although all of these global optimization idea are developed

under kriging models, we can apply them to ACOSSO with above empirical Bayesian viewpoint.

The augmented expected improvement criterion (Huang *et al.*, 2006b) is defined as

$$\mathsf{EI}(\boldsymbol{x}) = \mathbb{E}\left[\max\left\{\hat{f}(\boldsymbol{x}^{**}) - y(\boldsymbol{x}), 0\right\}\right] \cdot \left\{1 - \frac{\sigma}{\sqrt{s^2(\boldsymbol{x}) + \sigma^2}}\right\}$$

where the expectation is taken over the posterior distribution of $y(\boldsymbol{x})$ with $\hat{f}(\boldsymbol{x}^{**})$ kept fixed, $\boldsymbol{x}^{**}$ is the current 'effective best solution', $\hat{f}(\boldsymbol{x}^{**})$ is posterior mean of $y(\boldsymbol{x}^{**})$ and $s^2(\boldsymbol{x})$ is the posterior variance of $y(\boldsymbol{x})$. According to Huang *et al.* (2006b),

$$\boldsymbol{x}^{**} = \arg\max_{\boldsymbol{x}_1,\ldots,\boldsymbol{x}_n} u(\boldsymbol{x}),$$

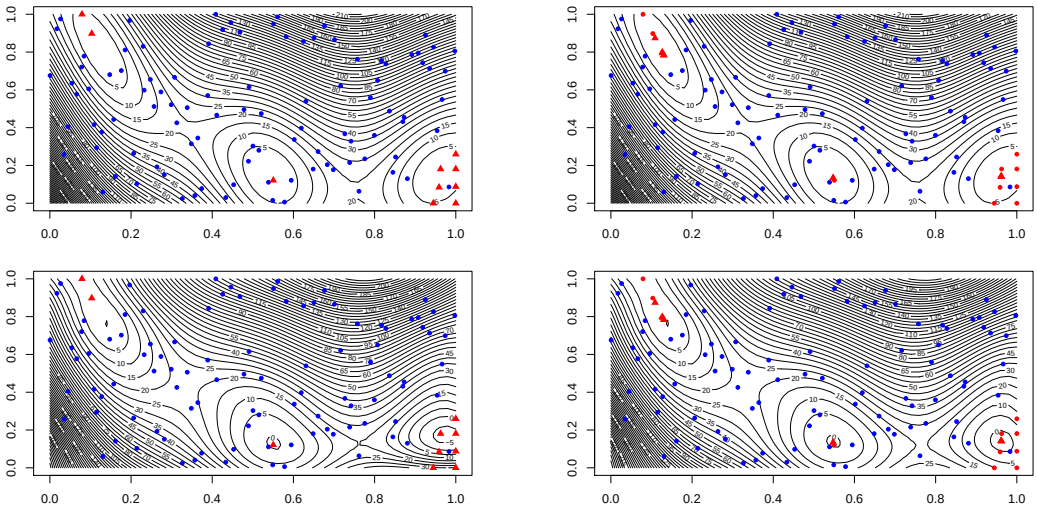
where $u(\boldsymbol{x}) = -\hat{f}(\boldsymbol{x}) - cs(\boldsymbol{x})$ is called the utility function and $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are the previously observed locations. Here, s is the posterior standard deviation of f and c is a tuning parameter, which reflects the degree of risk aversion. See Huang *et al.* (2006b) for more discussions. For our numerical illustrations, we choose c as the third quantile of a standard normal distribution. Under normality assumption, EI has a closed-form formulation:

$$\begin{aligned}\mathbb{E}\left[\max\left\{\hat{f}(\boldsymbol{x}^{**}) - y(\boldsymbol{x}), 0\right\}\right] = &\left\{\hat{f}(\boldsymbol{x}^{**}) - \hat{f}(\boldsymbol{x})\right\}\Phi\left\{\frac{\hat{f}(\boldsymbol{x}^{**}) - \hat{f}(\boldsymbol{x})}{s(\boldsymbol{x})}\right\} \\ &+ s(\boldsymbol{x})\phi\left\{\frac{\hat{f}(\boldsymbol{x}^{**}) - \hat{f}(\boldsymbol{x})}{s(\boldsymbol{x})}\right\},\end{aligned}$$

where Φ and ϕ are the standard normal cumulative distribution and normal probability density function respectively. For sequential batch design, we apply the constant liar algorithm (Ginsbourger *et al.*, 2010) with the augmented expected improvement criterion. An illustration of the sequential sampling is shown in Figure 3.2.

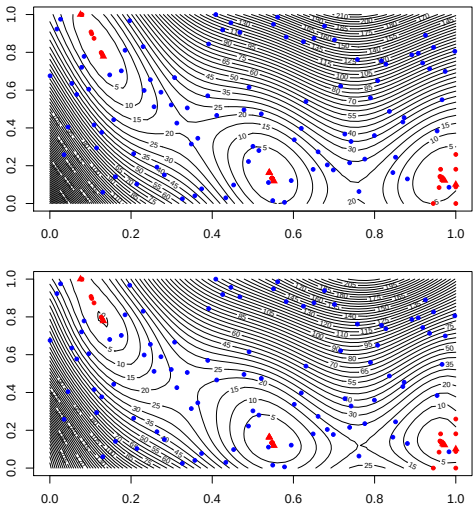
3.6 Credible set for minimizers

In this section we construct a credible set for minimizers. Although most of the procedure described in this section works for kriging, the computational time grows dramatically as dimensionality grows. Thus, it is usually computationally infeasible to

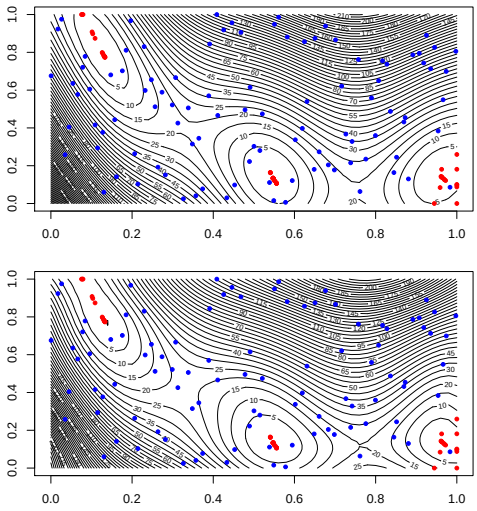


initial sample

sample 1



sample 3



sample 5

Figure 3.2. An illustration of the sequential sampling procedure (ACOSSO with Gaussian kernel). The setting is adapted from Section 3.7 (Branin function). The true function has three global minimizers and is of 13 dimensions with only 2 of them are useful. Blue dots: initial sampling locations. Red dots: proposed sampling locations so far (sampled). Red triangles: newly proposed sampling locations (not yet sampled).

apply the credible set idea for kriging models. With variable selection, our method can be used with the methodology developed in this section to obtain a credible set for the minimizers. Note that the idea of a credible set is extremely useful when the objective function has a few global optimizers or the value of global optimum is very close to values of some local optima. For instance, in our Branin function example, there are three global minimizers. It may turn out that all three minimizers are important and one does not want to miss any of them. Ordinary global optimization techniques tend to return one global minimizer without noticing the possibilities of multiple global minimizers. Similarly, if some local optima have very close value as the unique global optima does. Ordinary global optimization techniques may miss the global optima without any notice for other possible locations of global minimizers. The credible set serves as a safeguard in these situations.

Note that ACOSSO has the posterior interpretation as described in Section 3.3. And thus we can generate posterior sample from the posterior distribution and numerically estimate the posterior distribution of the optimizers. To be precise, we have the posterior distribution of a function and generating a sample of a function can be computationally expensive. One straightforward idea is to generate the values of the function over a dense grid. However, as the dimension grows, the number of grid points grow exponentially. And for estimating the optimizer of a posterior sample accurately, we need a fine grid. Thus, instead, we develop a fast and accurate algorithm for generating a sample of optimizer from the posterior distribution. We call this algorithm 'zoom-in' algorithm, as it evaluates a posterior sample adaptively through iteratively zooming into its minimizer. The 'zoom-in' algorithm for sampling a minimizer is given as follows.

1. Lay down a Latin Hypercube design of size N over the space of index set of f and generate sample over the design from the posterior distribution.

2. Find design points with K smallest sampled values.

3. Apply k-means clustering over those K design points and group them into L

groups.

4. For each group, lay down a smaller patch centering at the group mean and draw a new Latin Hypercube sample over this smaller patch, by conditioning on the existing samples. For speed, the new sample is drawn by conditioning on samples inside a relatively larger patch that contains the smaller patch.

5. For each group, find the minimizer.

6. Repeat (4) by laying down a even smaller patch with center of their current minimizer, for certain number of times.

7. Compare the current minimizers from each group and return the one with the minimum value.

The above algorithm is illustrated in Figure 3.4. It shows the sampling locations for a particular process generated from the posterior distribution. We generate the credible set for minimizers using the following algorithm:

1. Sample M minimizers using the above algorithm.

2. Apply kernel density estimation over these M minimizers.

3. Find the highest density set.

3.7 Simulation study

A simulation study is conducted to evaluate the practical performance of the proposed methodology. Here we use a popular test function in the optimization literatures. This function is called Branin function, which has two input variables. This function is defined over $[-5, 10] \times [0, 15]$ and we rescale its domain to $[0, 1] \times [0, 1]$ and depict it in Figure 3.3 (Left). Branin function is a challenging function to minimize as it has three global minima: $\boldsymbol{x}_1^* = (0.9616520, 0.15)^\intercal$ $\boldsymbol{x}_2^* = (0.1238946, 0.8166644)^\intercal$ and $\boldsymbol{x}_3^* = (0.5427730, 0.15)^\intercal$. To demonstrate the added difficulty of high dimensionality, we assume that there are 13 input dimensions and clearly only 2 of them are useful.

Moreover, we also tilt the Branin function so that the tilted version only has one global minimizer. See Figure 3.3 (Right). In the following, we provide a simulation study using both the Branin and tilted Branin function with the following detailed simulation setting:

- Total number of variables: $p = 13$
- Initial sample size: $n_0 = 100$
- Sequential sample size: $n_i = 10$ for $i = 1, \ldots, 5$
- Noise level, σ: 3 (Gaussian noise) (around 1% of the range of the Branin function)

And we compare the following procedures.

1. ss-acosso: Sequential sampling with ACOSSO with Sobolev's kernel.
2. ss-acosso-gauss: Sequential sampling with ACOSSO with Gaussian kernel.
3. ss-gp: Sequential sampling with kriging.
4. acosso: ACOSSO on augmented LHS.
5. acosso-gauss: ACOSSO with Gaussian kernel on augmented LHS.
6. gp: Kriging on augmented LHS.

For acosso, acosso-gauss and gp, the LHS is generated through augmenting the initial design and maintaining the Latin properties of the design (and attempting to add the points to the design in a way that maximizes S optimality). Thus, all methods share the same n_0 samples. All ACOSSO are fitted using weights from an initial COSSO fitting (Lin and Zhang, 2006) and both tuning parameters are selected by BIC. The numerical results are reported in Tables 3.1 and 3.2. Here, for all 500 simulations, all procedures obtain 100% superset selection (ss-gp and gp implement no variable selection). Generally, the sequential sampling procedures (ss-acosso, ss-acosso-gauss and ss-gp) perform better than their one-time sampling counterparts (acosso, acosso-gauss and gp). Those procedures with variable selection have better results than the

	ss-acosso	ss-acosso-gauss	ss-gp	acosso	acosso-gauss	gp
Correct selection	92.8%	100%	0%	77%	100%	0%
Superset selection	100%	100%	100%	100%	100%	100%
Minimum	4.66	0.728	9.15	8.30	6.10	9.99
Minimizer	0.0521	0.0407	0.0963	0.120	0.0901	0.0675
$\boldsymbol{x}_1^*$	0.252	0.386	0.336	0.678	0.578	0.366
$\boldsymbol{x}_2^*$	0.412	0.400	0.360	0.248	0.340	0.368
$\boldsymbol{x}_3^*$	0.336	0.214	0.304	0.074	0.082	0.266

Table 3.1. Simulation results: (1) Correct selection: proportion of selecting exactly all true predictors in the final fit. (2) Superset selection: proportion of selecting a superset of the true predictors in the final fit. (3) Minimum: The RMSE of the minimum. (4) Minimizer: The RMSE from the closet true minimizers. (5) $\boldsymbol{x}_i^*$: The proportion that the closest true minimizer is $\boldsymbol{x}_i^*$.

others. If we particularly focus on ss-gp and gp, we can see that the gain of sequential sampling is not much. It is probably because the computation and optimization of the EI is not numerically stable due to the high dimensionality.

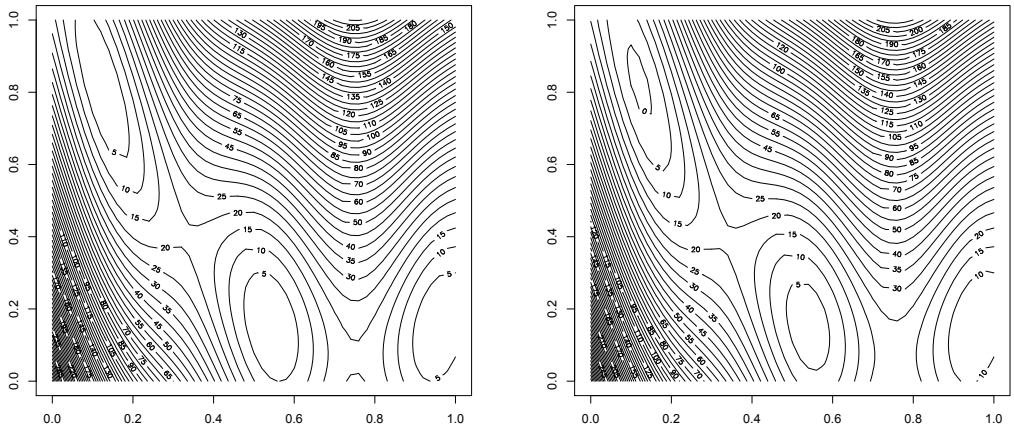


Figure 3.3. Left: The Branin function (rescaled). Right: The tilted Branin function (rescaled).

In addition, the 90%, 95% and 99% credible sets for ss-acosso-gauss in the above simulations is generated. And we summarize the performance of credible sets in Table 3.3. Also, illustrations of credible sets are given in Figure 3.5. Since this credible

	ss-acosso	ss-acosso-gauss	ss-gp	acosso	acosso-gauss	gp
Correct selection	93.6%	100%	0%	80.4%	99.8%	0%
Superset selection	100%	100%	100%	100%	100%	100%
Minimum	0.926	0.688	8.16	5.95	3.92	8.90
Minimizer	0.0682	0.0569	0.206	0.451	0.412	0.260

Table 3.2. Similar to Table 3.1, but for tilted Branin function.

	Branin function			Tilted Branin function	
Target coverage	Area	Coverage	Joint coverage	Area	Coverage
90%	0.0207	99.8%	40.8%	0.0077	95.2%
95%	0.0283	100%	56.6%	0.0101	98.2%
99%	0.0461	100%	83.6%	0.0155	99.8%

Table 3.3. (1) Area: the mean area of credible sets. (2) Coverage: the proportion that at least one of the three minimizers is included. (3) Joint coverage: the proportion that all three minimizers are included.

set is developed for one global minimizer, one needs a larger credible set to obtain the joint coverage. In practice, if one increases the level of the credible set, one usually see the set is formed by a few disjoint sets, e.g. in Figure 3.5, and this is a sign of multiple global optimizers or existence of local optimizers that have close function values as the global optimizer has.

3.8 Concluding remarks

In this work, we propose a global optimization technique for high dimensional expensive black-box functions. Our technique automatically selects useful variables (input dimensions) in each steps of the sequential sampling. This eases and stabilizes the optimization of the criterion used for designing the next sampling locations in the sequential sampling. As another contribution, we also provide a uncertainty quantification technique for the proposed optimization method. This technique provides a mean to compute credible sets for optimizer, which is very useful in global optimization for determining the quality of the estimated optimizer.

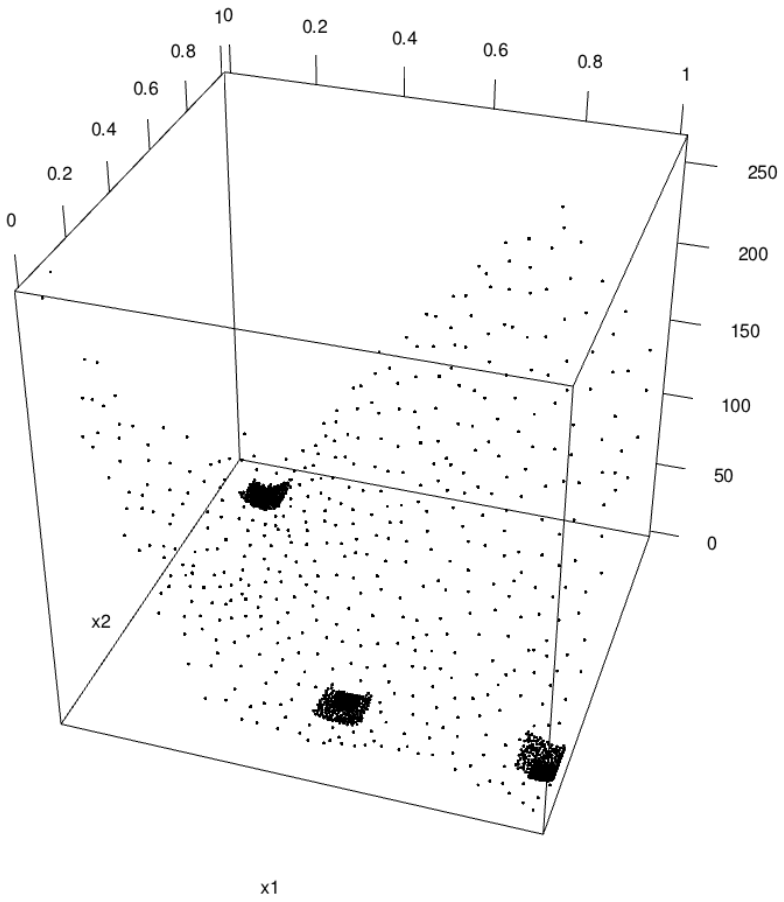


Figure 3.4. Illustration of the "zoom-in" algorithm.

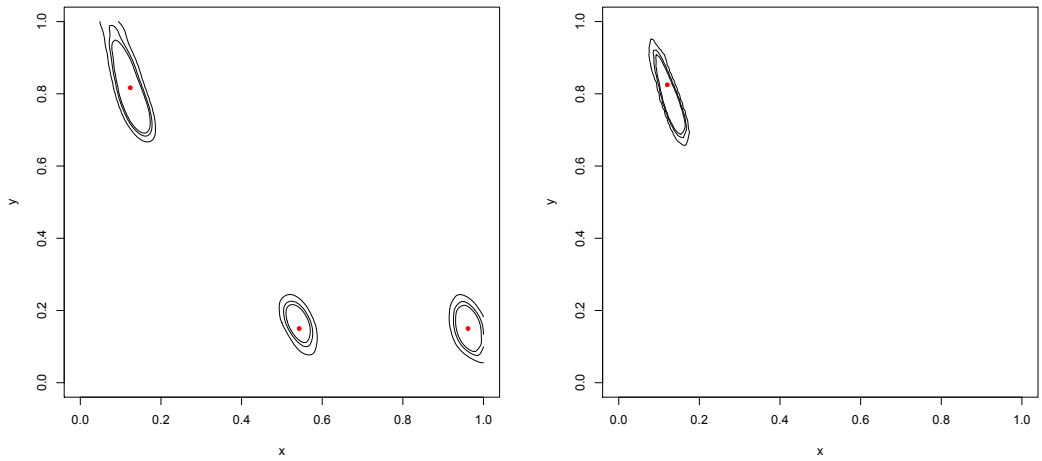


Figure 3.5. Illustration of the 90%, 95% and 99% credible sets. True minimizers are shown as red dots. Left: for Branin function. Right: for tilted Branin function.

Chapter 4

A Frequentist Approach to Computer Model Calibrations

Abstract

We consider the computer model calibration problem and provide a general frequentist approach with uncertainty quantification. Under the proposed framework, the data model is semi-parametric with a nonparametric discrepancy function to account for any discrepancy between the physical reality and the simulator. In an attempt to solve the fundamental identifiability issue between model parameters (of the simulator) and discrepancy function, we provide a new parametrization of the problem And for uncertainty quantification, a bootstrapping approach is used to provide a simple but effective method for construction of confidence regions of the quantities of interests.

This is a joint work with Curtis B. Storlie[1] and Thomas C. M. Lee[2].

4.1 Introduction

In many areas, complex mathematical models, implemented as computer codes, are used to model the physical reality. However, some computer models cannot be used for this goal without the specification of the values of some parameters, called model parameters. One would want to specify their values such that, with these values, the

[1]Los Alamos National Laboratory
[2]Department of Statistics, University of California at Davis

computer model can best reproduce the physical reality. The goal of calibration is to find such parameter values. In computer model calibration problem (Kennedy and O'Hagan, 2001), we observe an output y from the physical reality ζ at n locations of "controllable" input $\boldsymbol{x} = (x_1, \ldots, x_p)^\intercal$:

$$y_i = \zeta(\boldsymbol{x}_i) + \varepsilon_i,$$

for $i = 1, \ldots, n$. Here ε_i is the observation error for the i-th observation. The computer model $\eta(\boldsymbol{x}, \boldsymbol{\theta})$, also called the simulator, can be used to approximate the physical reality $\zeta(\boldsymbol{x})$ when the model parameter $\boldsymbol{\theta} = (\theta_1, \ldots, \theta_d)^\intercal$ is set to an approximate but unknown value $\boldsymbol{\theta}_0$. To account for the discrepancy between the simulator and the physical reality, one can assume there exists a discrepancy function $\delta_0(\boldsymbol{x})$ and the model of the experimental data can be written as

$$y_i = \eta(\boldsymbol{x}_i, \boldsymbol{\theta}_0) + \delta_0(\boldsymbol{x}_i) + \varepsilon_i, \tag{4.1}$$

for $i = 1, \ldots, n$. In order to estimate θ_0 and δ_0, we will need to be able to evaluate η at different values of $\boldsymbol{x}$ and $\boldsymbol{\theta}$. However, evaluation of $\eta(\boldsymbol{x}, \boldsymbol{\theta})$ is often very time expensive due to complex nature of the mathematical models. This complication facilitates the use of a surrogate model, referred to as an emulator (Higdon *et al.*, 2004; Kennedy and O'Hagan, 2001; Reich *et al.*, 2009), for the simulator. Typically a Gaussian process (GP) is assumed for the simulator to form a flexible emulator. Moreover, GP is often used to model the discrepancy function. To form the emulator, we obtain an additional set of observations from the simulator at m design locations $(\boldsymbol{x}_1^*, \boldsymbol{\theta}_1^*), \ldots, (\boldsymbol{x}_m^*, \boldsymbol{\theta}_m^*)$. Note that, we have observed two data sets thus far, with one from the physical reality and the other from the simulator. In this setting, we have to estimate $\boldsymbol{\theta}_0$, δ_0 and η.

Typically, the computer model calibrations are done within a Bayesian framework (Higdon *et al.*, 2004; Kennedy and O'Hagan, 2001; Reich *et al.*, 2009; Storlie *et al.*, 2013a, e.g.,). There have been frequentist approaches to the calibration problem, the most common of which involves obtaining the maximum likelihood estimator (MLE) for θ directly by evaluating η sequentially in an optimization routine (Huang *et al.*,

2006b; Jones *et al.*, 1998; Vecchia and Cooley, 1987), for example. Sometimes this may be too computationally prohibitive and a surrogate model (e.g., Storlie and Helton (2008); Storlie *et al.* (2009)) could be used in place of η for the purpose of obtaining an MLE for θ. However, this latter approach must be used with care as to not ignore the estimation error uncertainty in the surrogate model for η, which can often be substantial. Also, neither approach above allows for a model form discrepancy δ_0. While some models may be very good approximations, leading to a small discrepancy, no model is perfect and neglecting discrepancy is a major pitfall. However, the incorporation of the discrepancy function leads to an identifiability issue, which we will address in Section 4.2.

Because of the GP's ability to incorporate uncertainty about the surrogate model for $\eta(\boldsymbol{x}, \boldsymbol{\theta})$, and a similar ability to provide uncertainty about the discrepancy function δ_0, the Bayesian approach has been prominent in computer model calibration problems, particularly when η is expensive to evaluate. However, we are interested here in solving this problem from a purely frequentist perspective, while also accounting for uncertainty in the model parameters, surrogate model, and discrepancy. To the best of our knowledge, there has been no previous attempt to solve the computer model calibration problem in this manner. Yet, there are several reasons for doing this: (i) The proposed approach is simple, robust, and easy to understand, and it will work with any choice of surrogate model. (ii) It provides a complementary calibration result to the Bayesian approach, each approach providing some qualitative confirmation of the other. (iii) Many researchers in computational physics are opposed to the Bayesian calibration approach because of the complex prior assumptions of GP and identifiability concerns between emulator and discrepancy. The proposed approach now provides a viable alternative that now accounts for potentially important sources of uncertainty. While any statistical model makes assumptions, there are fewer assumptions necessary in the proposed approach than in the Bayesian approach. For example, prior distributions on the emulator and discrepancy are replaced with "smoothness" assumptions and an emphasis is intuitively placed on the

ability to predict the experimental data via cross validation. Empirical evidence shows that our proposed frequestist approach tends to give better performance than existing Bayesian approaches.

4.2 A semi-parametric modeling to calibration problem

In this section we aim to develop a frequentist approach to calibration problems. Consider the semi-parametric model (4.1) for the physical reality. Despite its popularity under Bayesian framework, this model is not identifiable in the frequentist regime, where $\boldsymbol{\theta}_0$ and δ_0 are treated as fixed. In the following, we provide intuitive and identifiable definition for $\boldsymbol{\theta}_0$ and δ_0 under model (4.1). For ease of exploration, a general framework for the estimation of $\boldsymbol{\theta}_0$ and δ_0 is proposed under the knowledge of η in Section 4.2.1 and 4.2.2. We delay the discussion of emulator of η to Section 4.2.3. The proposed framework takes advantage of existing optimization techniques and nonparametric regression techniques, and thus provide an effective and flexible approach to estimate $\boldsymbol{\theta}_0$ and δ_0 with easy implementation.

4.2.1 Identifiability issue

Under frequentist interpretation, $\boldsymbol{\theta}_0$ and δ_0 are treated as fixed values. In model (4.1), the discrepancy function δ_0 is usually assumed to be an unconstrained smooth function, where nonparametric regression techniques are commonly used to estimate such function. To see the non-identifiability of (4.1), imagine there are two different values of $\boldsymbol{\theta}$, say $\boldsymbol{\theta}_1$ and $\boldsymbol{\theta}_2$. Now, we can write $\delta_1(\boldsymbol{x}) = \zeta(\boldsymbol{x}) - \eta(\boldsymbol{x}, \boldsymbol{\theta}_1)$ and $\delta_2(\boldsymbol{x}) = \zeta(\boldsymbol{x}) - \eta(\boldsymbol{x}, \boldsymbol{\theta}_2)$. Here $(\boldsymbol{\theta}_1, \delta_1)$ and $(\boldsymbol{\theta}_2, \delta_2)$ both give the same distribution of y in model (4.1) and this leads to an identifiability issue. Even though prior information or penalty can be incorporated in the estimation procedure to "bias" the estimator towards certain values, the fundamental identifiability issue in the above model still exists. This issue prevents us from defining the $\boldsymbol{\theta}_0$ and δ_0 even one knows ζ completely. This identifiability issue forbids us from constructing uncertainty measure such as confidence intervals of $\boldsymbol{\theta}_0$ or confidence bands of δ_0, since there is no well-defined target parameters.

Here we attempt to give sensible definitions of $\boldsymbol{\theta}_0$ and δ_0 in order to achieve identifiable modeling. Write the spaces of model parameters and inputs as $\boldsymbol{\Theta}$ and $\mathcal{X}$. We propose the following model of the physical reality ζ:

$$\zeta(\boldsymbol{x}) = \eta(\boldsymbol{x}, \boldsymbol{\theta}_0) + \delta_0(\boldsymbol{x}),$$

where η is the simulator (a smooth function), $\boldsymbol{\theta}_0 = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} \int_{\mathcal{X}} (\zeta(\boldsymbol{x}) - \eta(\boldsymbol{x}, \boldsymbol{\theta}))^2 \, dF(\boldsymbol{x})$, δ is an unknown smooth function (discrepancy) and F characterizes a weighing scheme of $\boldsymbol{x}$. F is related to the sampling design and is usually an identity function for common sampling design schemes. Under an regularity assumption (Assumption 4.6 in Section 4.4), both $\boldsymbol{\theta}_0$ and δ_0 are identifiable. Clearly, there exists different ways to define $\boldsymbol{\theta}_0$ and δ_0. We choose these definitions as they match with the intuition that $\boldsymbol{\theta}_0$ is the most plausible one, i.e. the one makes η closest to ζ, and δ is for taking care of the left-over.

4.2.2 Estimation

Now, suppose we observe the physical system ζ at n locations $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$, i.e. $y_i = \zeta(\boldsymbol{x}_i) + \varepsilon_i$, $i = 1, \ldots, n$, where ε_i is the observation error for the i-th observation. These errors are assumed to be independent and have mean 0. For simplicity, we assume $\mathcal{X} = [0,1]^p$, where p is the number of inputs. With the above modeling of ζ, the observations are assumed to follow

$$y_i = \eta(\boldsymbol{x}_i, \boldsymbol{\theta}_0) + \delta_0(\boldsymbol{x}_i) + \varepsilon_i. \tag{4.2}$$

Here we consider a simpler situation by assuming η is known. In calibration problems, we can estimate η by a second set of samples, which we will describe later in details in Section 4.2.3. The definitions of $\boldsymbol{\theta}_0$ and δ_0 motivate a two-step procedure for their estimation:

1. (Optimization) Compute the estimate of $\boldsymbol{\theta}$, $\hat{\boldsymbol{\theta}} = \arg\min_{\boldsymbol{\theta}\in\boldsymbol{\Theta}} M_n(\boldsymbol{\theta})$ where

$$M_n(\boldsymbol{\theta}) = \frac{1}{n} \sum_{i=1}^{n} \{y_i - \eta(\boldsymbol{x}_i, \boldsymbol{\theta})\}^2.$$

2. (Nonparametric regression) Estimate δ via common nonparametric regression on $\{(\boldsymbol{x}_i, y_i - \eta(\boldsymbol{x}_i, \hat{\boldsymbol{\theta}}))\}_{i=1}^n$.

This estimation strategy has its beauty in flexibility and ease of implementation. This can be coupled with different (global) optimization techniques and nonparametric regression methods. Since the estimation of $\boldsymbol{\theta}_0$ and that of δ_0 are separated, one does not have to worry about re-running the optimization to choosing the smoothing parameter in the nonparametric regression. In general, this strategy is very efficient in computation. For numerical illustrations in this paper, we adopt the genetic optimization using derivative (Sekhon and Mebane, 1998) for the (global) optimization of $\boldsymbol{\theta}$ and smoothing spline ANOVA (Wahba, 1990) for the nonparametric regression. The theoretical results of these estimators are also provided in Section (4.4) under fixed design setting.

4.2.3 Emulator

Since the simulator η is expensive to run, a common approach is to use a surrogate model, also called an emulator. The surrogate model is typically an nonparametric regression model that is estimated via a second set of samples of the simulator. Let the simulator output at several (m) design locations $(\boldsymbol{x}_1^*, \boldsymbol{\theta}_1^*), \ldots, (\boldsymbol{x}_m^*, \boldsymbol{\theta}_m^*)$ be denoted $\boldsymbol{y}_s = (y_{1,s}, \ldots, y_{m,s})^\intercal$. They are assumed to follow:

$$y_{s,j} = \eta(\boldsymbol{x}_j^*, \boldsymbol{\theta}_j^*) + \tau_j, \qquad \forall j = 1, \ldots, m,$$

where τ_j's are independent random errors with mean zero. At a high level, the proposed approach works as follows. We use the evaluations of the simulator $\boldsymbol{y}_s$ to fit a surrogate model via a nonparametric regression such as SS-ANOVA (Wahba, 1990) and ACOSSO (Storlie *et al.*, 2011). We then treat the surrogate $\hat{\eta}$ as fixed in a semiparametric regression problem (4.2) to estimate $\boldsymbol{\theta}_0$ and δ_0 via methodology described in Section 4.2.2. The parameters $\boldsymbol{\theta}$ can be constrained to a particular domain, as is often done in the Bayesian calibration approach via a prior distribution. Notice that the estimation of η and δ_0 is done separately. To the best of our knowledge, this approach to obtain a point estimate for the calibration problem with discrepancy of a

computationally demanding model has not yet been attempted until now. The above description does not yet account for the uncertainty in the estimation of the surrogate, model parameters, or the discrepancy. However, this issue can easily be addressed via bootstrap sampling (see, e.g., Davison, 1997; Efron and Tibshirani, 1994), as described in Section 4.3.

4.3 A bootstrapping approach to uncertainty quantification

Bootstrap sampling was used successfully to address the uncertainty in the surrogate model for the purpose of sensitivity analysis (SA) and uncertainty analysis (UA) of computationally demanding models in Reich *et al.* (2009) and Storlie *et al.* (2013b). The calibration problem is far more complicated than SA/UA due to the estimation of $\boldsymbol{\theta}_0$ and δ_0, but a very similar approach can be applied to the calibration estimate proposed above.

Let the point estimates of the unknown parameters in the data model in (4.1) be obtained as described above and be denoted $\hat{\boldsymbol{\theta}}$, $\hat{\eta}$, and $\hat{\delta}$. These define an estimate for the data generating process for both the simulator data $\boldsymbol{y}_s$ and experimental data $\boldsymbol{y}$. One may re-sample the designs in both data sets if the data are generated under random designs. Thus, we can produce B bootstrap samples by re-sampling (centered) residuals and re-estimate the parameters to obtain B bootstrap estimates of $\boldsymbol{\theta}$, η, and δ, denote them $\hat{\boldsymbol{\theta}}_b^*$, $\hat{\eta}_b^*$, and $\hat{\delta}_b^*$, $b = 1, \ldots, B$, respectively. This bootstrap sample of estimates can be used to obtain a bootstrap confidence region for most quantities of interest. As in calibration problem, confidence intervals for elements of $\boldsymbol{\theta}_0$ and (pointwise) confidence band for δ_0 are usually interested. For example, if we wish to obtain a confidence interval for $\theta_{0,1}$, the first element of $\boldsymbol{\theta}_0$, then we can do so by finding the $\alpha/2$ and $(1-\alpha/2)$ sample quantiles from the collection $\{\hat{\theta}_{1,1}^*, \ldots, \hat{\theta}_{B,1}^*\}$, where $\hat{\theta}_{b,1}^*$ represents the first element of $\hat{\boldsymbol{\theta}}_b^*$ for $b = 1, \ldots, B$, and write them as $z_{\alpha/2}^*$ and $z_{1-\alpha/2}^*$, respectively. The corresponding confidence interval is then given by $(z_{\alpha/2}^*\ ,\ z_{1-\alpha/2}^*)$. A confidence interval for a prediction of the physical system ζ at a new input $\boldsymbol{x}_{\text{new}}$

and the pointwise confidence band for δ_0 can be obtained in a similar fashion.

Since our procedure involves nonparametric regression, the impact of bias may lead to incorrect asymptotic coverage of the aforementioned bootstrap confidence regions (see, e.g., Hall, 1992a,b; Härdle and Bowman, 1988). In the literature, there are two common strategies for correcting the coverage: undersmoothing and oversmoothing. As shown in Hall (1992a), undersmoothing is a simpler and more effective procedure than oversmoothing. Thus, we can modify the above procedure by incorporating some undersmoothing (i.e. choosing a smaller smoothing parameter than what is chosen by cross-validation). However, it is not uncommon that the issue of bias is completely avoided, which results in the use of the non-adjusted confidence regions desribed above. See, e.g., Efron and Tibshirani (1994) and Ruppert *et al.* (2003).

4.4 Theoretical results

Note that the estimation of η depends on the second independent sample from the simulator of size m and, in practice, we usually have larger sample from the simulators than the one from physical reality. Thus, it is reasonable to assume that m is of a higher order than n to go to infinity in the asymptotic framework. If m is fast enough, the asymptotics of $\hat{\theta}$ and $\hat{\delta}$ are similar to those under known η. For simplicity, here we assume that η is known and derive the asymptotic property of $\hat{\theta}$ and $\hat{\delta}$ described in Section 4.2.2.

Write $\hat{\boldsymbol{\theta}}$ and $\hat{\delta}$ as $\hat{\boldsymbol{\theta}}_n$ and $\hat{\delta}_n$ respectively to address their dependence on n. In the following, we assume that $\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n$ are fixed and write $F_n = \sum_{i=1}^n \delta_{\boldsymbol{x}_i}/n$. In addition, $\|\cdot\|_n$ represents the $L_2(F_n)$-norm and, with slight abuse of notations, $\|\cdot\|$ represents both the $L_2(F)$-norm and the Euclidean norm. The context should be clear enough for correct interpretation. For two functions g and h, let $\langle g, h\rangle_n = \sum_{i=1}^n g(\boldsymbol{x}_i)h(\boldsymbol{x}_i)$ and $\langle g, h\rangle = \int_{\mathcal{X}} g(\boldsymbol{x})h(\boldsymbol{x})dF(\boldsymbol{x})$. With slight notation abuse, we also write $\langle y, g\rangle_n = (1/n)\sum_{i=1}^n y_i g(\boldsymbol{x}_i)$ and $\langle \varepsilon, g\rangle_n = (1/n)\sum_{i=1}^n \varepsilon_i g(\boldsymbol{x}_i)$. We also write $g_{\boldsymbol{\theta}}(\boldsymbol{x}) = \eta(\boldsymbol{x}, \boldsymbol{\theta})$, $\mathcal{G} = \{g_{\boldsymbol{\theta}} : \boldsymbol{\theta} \in \Theta\}$ and $\mathcal{G} - g = \{g_{\boldsymbol{\theta}} - g : \boldsymbol{\theta} \in \Theta\}$ for a function g.

In practice, the design is usually either fixed or correlated (e.g. Latin Hypercube

sampling). Thus, our results are developed under fixed design, rather than the relatively common asymptotic framework of IID design. We first approach the parametric part and establish the $\sqrt{n}$-consistency of $\hat{\boldsymbol{\theta}}_n$ (Theorem 4.1), where the difficulty lies in the existence of the discrepancy. The effect is similar to a regression model with misspecification.

As for the discrepancy function, we adopt the framework of Section 10.1 of Van De Geer (2000) for penalized least squares estimation. We extend Theorem 10.2 of Van De Geer (2000) to obtain asymptotic behavior of $\hat{\delta}_n$ (Lemma B.2 and Theorem 4.2), under the effect of estimation error of $\hat{\boldsymbol{\theta}}_n$. Let the class of functions that δ_0 belongs be $\mathcal{H}$. Under the penalized least square framework, the general form of the estimate of discrepancy function is

$$\hat{\delta}_n = \arg\min_{\delta \in \mathcal{H}} \left(\frac{1}{n} \sum_{i=1}^{n} (y_i - g_{\hat{\boldsymbol{\theta}}_n}(\boldsymbol{x}_i) - \delta(\boldsymbol{x}_i))^2 + \lambda_n^2 J^v(\delta) \right), \tag{4.3}$$

where $v > 0$, $\lambda_n > 0$, $J : \mathcal{H} \to [0, \infty)$ is a pseudo-norm on $\mathcal{H}$. The λ_n is known as smoothing parameter.

As an illustration, we provide the convergence rate of $\hat{\delta}_n$ for $p = 1$ if penalized smoothing spline is used (Corollary 4.1). This requires an additional orthogonality argument for the application of Theorem 4.2. Note that we write $\boldsymbol{x}$ as x for $p = 1$.

Here are some assumptions:

Assumption 4.1 (Error structure)**.** $\mathbb{E}(\varepsilon_i) = 0$, $E(\varepsilon_i^2) = \sigma^2$ *for all* $i = 1, \ldots, n$. *Also,* $\varepsilon_1, \ldots, \varepsilon_n$ *are uniformly sub-Gaussian: There exists* K *and* σ_0 *such that*

$$\max_{i=1,\ldots,n} K^2 \left\{ \mathbb{E} \exp(\varepsilon_i^2 / K^2) - 1 \right\} \leq \sigma_0^2.$$

Assumption 4.2 (Parameter space)**.** Θ *is a totally bounded* d*-dimensional Euclidean space. That means, there exists* $R_1 > 0$ *such that* $\Theta \subset \mathcal{B}(R_1)$.

Assumption 4.3 (Function class $\mathcal{G}$)**.**

(a) There exists $c_0 > 0$ *such that* $\|g_{\boldsymbol{\theta}} - g_{\boldsymbol{\theta}'}\|_n \leq c_0 \|\boldsymbol{\theta} - \boldsymbol{\theta}'\|$ *for all* $\boldsymbol{\theta}, \boldsymbol{\theta}' \in \Theta$.

(b) $g_{\boldsymbol{\theta}}$ *is twice continuously differentiable with respect to* $\boldsymbol{\theta}$ *in a neighborhood of* $\boldsymbol{\theta}_0$. $g_{\boldsymbol{\theta}}^{(1)}(\boldsymbol{x})$ *and* $g_{\boldsymbol{\theta}}^{(2)}(\boldsymbol{x})$ *are continuous with respect to* $\boldsymbol{x}$ *over this neighborhood.*

(c) $\sup_{\boldsymbol{x}\in\mathcal{X}} |g_{\boldsymbol{\theta}}^{(1)}(\boldsymbol{x})|$ *and* $\sup_{\boldsymbol{x}\in\mathcal{X}} |g_{\boldsymbol{\theta}}^{(2)}(\boldsymbol{x})|$ *are bounded uniformly over a neighborhood of* $\boldsymbol{\theta}_0$.

Assumption 4.4.

(a) $\sup_{h\in(\mathcal{G}-\zeta)} \|h\|_n < \infty$.

(b) $\sup_{h\in(\mathcal{G}-g_{\boldsymbol{\theta}_0})} \|h\|_n < \infty$.

Assumption 4.5 (Convergence of design)**.**

(a) $\sup_{\boldsymbol{\theta}\in\Theta} |\|\zeta - g_{\boldsymbol{\theta}}\|_n^2 - \|\zeta - g_{\boldsymbol{\theta}}\|^2| = o(1)$.

(b) $\sup_{g\in\{(g_{\boldsymbol{\theta}}-g_{\boldsymbol{\theta}_0})/\|g_{\boldsymbol{\theta}}-g_{\boldsymbol{\theta}_0}\|_n:\boldsymbol{\theta}\in\Theta\}} |\langle \zeta - g_{\boldsymbol{\theta}_0}, g\rangle_n - \langle \zeta - g_{\boldsymbol{\theta}_0}, g\rangle| = \mathcal{O}(n^{-1/2})$.

(c) Elements of $|(1/n)\sum_{i=1}^n g_{\boldsymbol{\theta}_0}^{(1)}(\boldsymbol{x}_i) g_{\boldsymbol{\theta}_0}^{(1)}(\boldsymbol{x}_i)^\intercal - \int_{\mathcal{X}} g_{\boldsymbol{\theta}_0}^{(1)}(\boldsymbol{x}) g_{\boldsymbol{\theta}_0}^{(1)}(\boldsymbol{x})^\intercal dF(\boldsymbol{x})|$ *are* $o(1)$.

Assumption 4.6 (Identification)**.** *For all* $\epsilon > 0$, $\inf_{\boldsymbol{\theta}\in\Theta:\|\boldsymbol{\theta}-\boldsymbol{\theta}_0\|>\epsilon} \|\zeta - g_{\boldsymbol{\theta}}\| > \|\zeta - g_{\boldsymbol{\theta}_0}\|$.

Assumption 4.7 (Discrepancy function)**.**

(a) δ_0 *is continuous.*

(b) There exist $K > 0$ *and* $\alpha > 0$ *such that*

$$H\left(u, \left\{\frac{\delta - \delta_0}{J(\delta) + J(\delta_0)} : \delta \in \mathcal{H}, J(\delta) + J(\delta_0) > 0\right\}, F_n\right) \leq K\delta^\alpha,$$

for all $u > 0$ *and* $n \geq 1$.

The following are two theorems and a corollary. Their proofs can be found in Appendix B.

Theorem 4.1 (Rates of convergence of $\hat{\boldsymbol{\theta}}_n$ and $g_{\hat{\boldsymbol{\theta}}_n}$)**.** *Assume that Assumptions 4.1, 4.2, 4.3(a-c), 4.4, 4.5(a-c), 4.6, 4.7(a) hold. We have* $\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| = \mathcal{O}_p(n^{-1/2})$ *and* $\|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n = \mathcal{O}_p(n^{-1/2})$.

Theorem 4.2 (Rate of convergence of $\hat{\delta}_n$)**.** *Assume that conditions of Theorem 4.1 and Assumption 4.7(b) hold. Suppose* $v > (2\alpha)/(2+\alpha)$ *and* $\lambda_n \asymp n^{-1/(2+\alpha)}$.

(i) If $J(\delta_0) > 0$, *we have*

$$\|\hat{\delta}_n - \delta_0\|_n = \mathcal{O}_p\left(n^{-1/(2+\alpha)}\right).$$

(ii) If $J(\delta_0) = 0$, $J(\delta) > 0$ *for all* $\delta \in \mathcal{H}$, *and* $4v < (2+\alpha)(2v - 2\alpha + v\alpha)$, *we have*

$$\|\hat{\delta}_n - \delta_0\|_n = \mathcal{O}_p\left(n^{-1/2}\right).$$

Corollary 4.1 (Penalized smoothing spline)**.** *Assume* $p = 1$, $\mathcal{H} = \{\delta : [0,1] \to \mathbb{R}, \int_0^1 \{\delta^{(m)}(x)\}^2 dx < \infty\}$ *and* $J(\delta) = [\int_0^1 \{\delta^{(m)}(x)\}^2 dx]^{1/2}$. *And* $\hat{\delta}_n$ *is given in (4.3) with* $v = 2$. *Assume the conditions of Theorem 4.1 hold. Let* $\psi = (\psi_1, \ldots, \psi_k)^\intercal$, *where* ψ_k*'s are defined in (B.9). Assume that the smallest eigenvalue of* $\int \psi\psi^\intercal dF_n$ *is bounded away from 0. In addition, suppose* $\lambda_n \asymp n^{-1/(2+\alpha)}$.

(i) If $J(\delta_0) > 0$, *we have*

$$\|\tilde{\delta}_n - \delta_0\|_n = \mathcal{O}_p\left(n^{-m/(2m+1)}\right).$$

(ii) If $J(\delta_0) = 0$, *we have*

$$\|\tilde{\delta}_n - \delta_0\|_n = \mathcal{O}_p\left(n^{-1/2}\right).$$

4.5 Simulation study

We conducted a simulation study to investigate the practical performance of the proposed methodology. The simulation settings are as follows. We set $n = 50$ and $m = 300$. The input and parameter spaces are $\mathcal{X} = [0,1]$ and $\Theta = [0, 0.25] \times [0, 0.5]$ respectively. Both ε_i's and τ_j's follow normal distribution with standard deviation set as 0.741 and 0.906 respectively, where the signal-to-noise ratios are approximately equal to 10 and 55 respectively in the sampling of physical reality and that of simulator. Both designs in the simulator data and the experimental data are generated by Latin hypercube sampling. The simulator, parameter values and discrepancy function are

given as follows:

$$\begin{aligned}\eta(x,\boldsymbol{\theta}) &= 7\{\sin(2\pi\theta_1-\pi)\}^2+2\{2\pi\theta_2-\pi)^2\sin(2\pi x-\pi)\}\\ \boldsymbol{\theta}_0 &= (0.2,0.3)^\intercal\\ \delta_0(x) &= \cos(2\pi x-\pi)\end{aligned}$$

Here 200 data sets are simulated. The proposed calibration method is applied to each of these data sets. We use the smoothing spline ANOVA as the nonparametric regression model for both η and δ_0. The smoothing parameters are selected by generalized cross-validation (GCV). For uncertainty quantification, the following methods are applied.

1. fcal-smooth: Proposed method with undersmoothing, but not design re-sampling
2. fcal-smooth-sam: Proposed method with undersmoothing and design re-sampling
3. fcal: Proposed method without undersmoothing and design re-sampling
4. fcal-sam: Proposed method without undersmoothing, but with design re-sampling
5. bssanova: Calibration of computational models via Bayesian smoothing spline ANOVA (Storlie *et al.*, 2014)

For all methods with undersmoothing, we choose the smoothing parameter as 0.9 times the smoothing parameter selected by GCV.

The mean square errors (MSEs) of $\theta_{0,1}$ and $\theta_{0,2}$, where $\boldsymbol{\theta}_0 = (\theta_{0,1},\theta_{0,2})^\intercal$, are 1.614×10^{-4} (3.301×10^{-5}) and 7.069×10^{-5} (8.749×10^{-6}) respectively with standard errors shown in the parentheses. That means the corresponding root MSEs are 1.27×10^{-2} and 8.41×10^{-3}, which are small compared to the true parameter value. For the discrepancy, the MSE over a fine grid is 0.09428 (0.005247), with standard error shown in the parenthesis. As for uncertainty quantification, the simulation results are summarized in Table 4.1. The Bayesian smoothing spline ANOVA method bssanova can be used for comparison. Overall, Table 4.1 shows that our proposed methods performs

better than bssanova. In addition, the results of pointwise confidence (credible) intervals for δ_0 are summarized in Figure 4.1. This also shows that the proposed methods perform better than bssanova.

Table 4.1. Simulation results of 95% confidence (credible) intervals of $\boldsymbol{\theta}_0$: Average coverages and lengths of 95% confidence (credible) intervals. The standard errors are shown in parentheses.

	fcal-smooth	fcal-smooth-sam	fcal	fcal-sam	bssanova
coverage	98.5% (0.862%)	97.5 (1.11%)	9.75e-01 (1.11%)	98.5% (0.862%)	88.5% (2.26%)
	94.0% (1.68%)	96.0% (1.39%)	94.5% (1.62%)	96.0% (1.39%)	95.5% (1.47%)
length	4.23e-02 (1.07e-03)	4.18e-02 (1.01e-03)	4.07e-02 (1.05e-03)	4.10e-02 (1.02e-03)	1.19e-01 (2.74e-03)
	3.18e-02 (3.74e-04)	3.20e-02 (3.68e-04)	3.17e-02 (3.45e-04)	3.14e-02 (3.39e-04)	4.32e-02 (4.11e-04)

4.6 Concluding remarks

In this work, we provide a frequentist framework for computer model calibration. This framework applies a general semi-parametric data model with discrepancy function, which allows discrepancy between simulator and the physical reality. Despite the flexibility of the model, our proposed framework gives identifiable parametrizations for both the model parameters and the discrepancy function. These parametrizations matches with the general belief of the roles of the model parameters and the discrepancy function. Simple but effective algorithm has been proposed for estimation. In addition, we provide theoretical results for the proposed calibration approach. Our work also provides a bootstrapping approach for uncertainty quantification. Due to simplicity of the proposed calibration framework and the corresponding bootstrap, our approach can be coupled with variety of optimization methods and emulators, which is beneficial to practitioners.

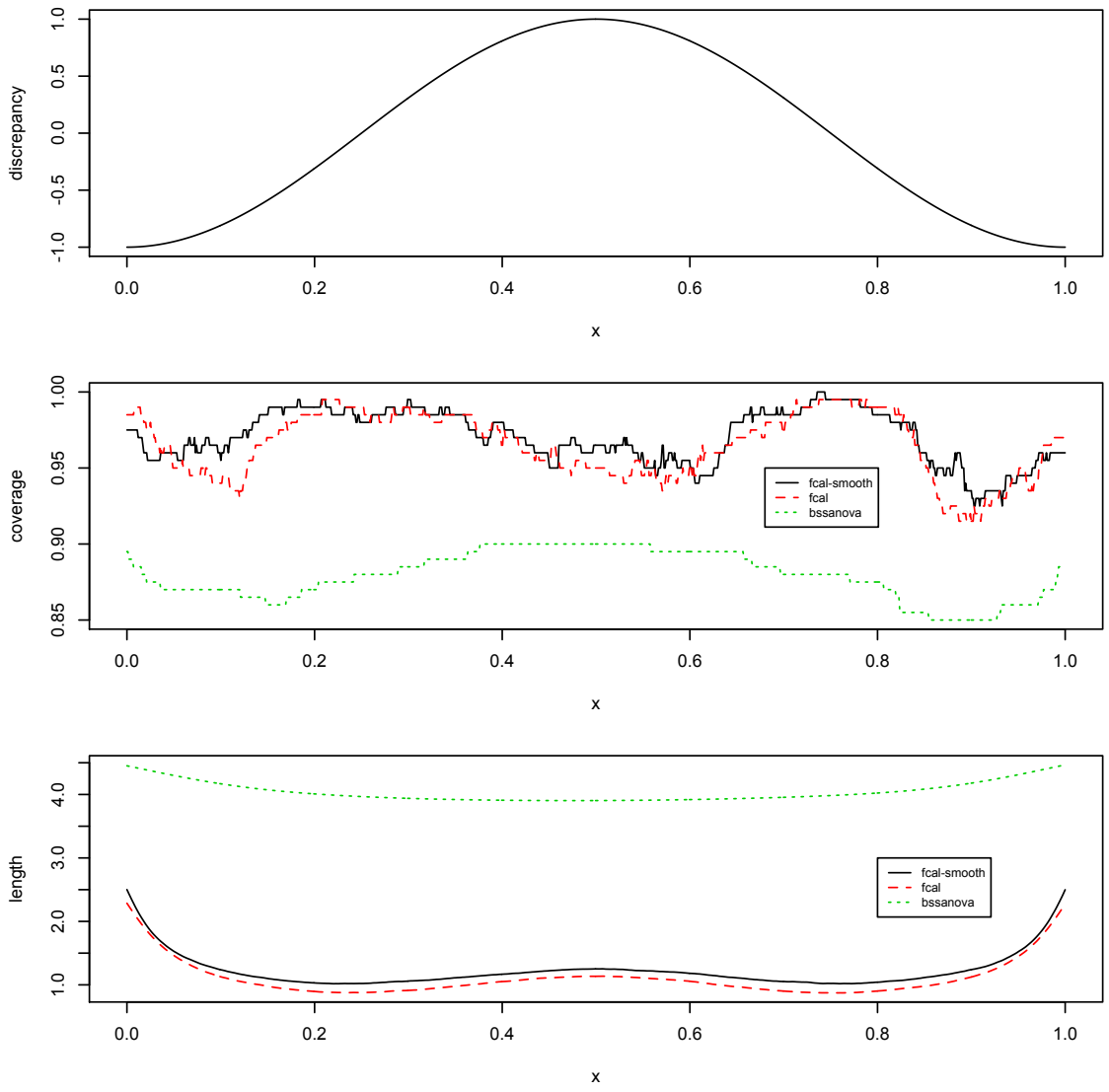


Figure 4.1. Simulation results of 95% pointwise confidence (credible) bands of δ_0: Top: True discrepancy function. Middle: Average coverage. Bottom: Average length.

Chapter 5

Automatic Estimation of Flux Distributions of Astrophysical Source Populations

Abstract

In astrophysics a common goal is to infer the flux distribution of populations of scientifically interesting objects such as pulsars or supernovae. In practice, inference for the flux distribution is often conducted using the cumulative distribution of the number of sources detected at a given sensitivity. The resulting "$\log(N > S) - \log(S)$" relationship can be used to compare and evaluate theoretical models for source populations and their evolution. Under restrictive assumptions the relationship should be linear. In practice, however, when simple theoretical models fail, it is common for astrophysicists to use pre-specified piecewise linear models. This paper proposes a methodology for estimating both the number and locations of "breakpoints" in astrophysical source populations that extends beyond existing work in this field.

An important component of the proposed methodology is a new Interwoven EM Algorithm that computes parameter estimates. It is shown that in simple settings such estimates are asymptotically consistent despite the complex nature of the parameter space. Through simulation studies it is demonstrated that the proposed methodology is capable of accurately detecting structural breaks in a vari-

ety of parameter configurations. This paper concludes with an application of our methodology to the *Chandra* Deep Field North (CDFN) dataset.

This is a joint work with Paul Baines[1], Alexander Aue[1], Thomas C. M. Lee[1] and Vinay L. Kashyap[2]. This work will appear in the Annals of Applied Statistics.

5.1 Introduction

The relationship between the number of sources and the threshold at which they can be detected is an important tool in astrophysics for describing and investigating the properties of various types of source populations. Known as the $\log N - \log S$ relationship, the idea is to use the number of sources $N(> S)$ that can be detected at a given sensitivity level S, on the log-log scale, to describe the distribution of source fluxes. In simple settings and under restrictive assumptions a linear relationship between the log-flux and the log-survival function can be derived from first principles. Traditionally astrophysicists have therefore examined this relationship by characterizing the slope of the log of the empirical survival function as a function of the log-flux of the sources.

One of the first examples of the $\log N - \log S$ relationship being derived from first principles is in Scheuer (1957). It is shown that if radio stars are uniformly distributed in space then the number with intensity exceeding a threshold S is given by $N(> S) \propto S^{-3/2}$. Importantly, the relationship holds irrespective of several factors such as luminosity dispersion and the reception pattern of the detector. The derived relationships therefore allow for researchers to test for departures from specific theories. For example, Hewish (1961) uses the derived relationship to infer a non-uniform distribution of sources for a particular population.

Other examples of $\log N - \log S$ analyses include Guetta *et al.* (2005), who use the relationship for Gamma Ray Bursts (GRBs) to constrain the structure of GRB jets. By comparing the $\log N - \log S$ relationship for observed data to the predicted $\log N - \log S$ relationship under different physical models for GRB jets, the authors

[1]Department of Statistics, University of California at Davis
[2]Harvard-Smithsonian Center for Astrophysics

are able to uncover limitations in the physical models. The $\log N - \log S$ curves have also been used to constrain cosmological parameters using cluster number counts in different passbands; see, e.g., Mathiesen and Evrard (1998) and Kitayama *et al.* (1998). Other applications of $\log N - \log S$ modeling include the study of active galactic nuclei (AGNs). For example, Mateos *et al.* (2008) use the $\log N - \log S$ relationship over different X-ray bands to constrain the population characteristics of hard X-ray sources.

Under independent sampling, the linear $\log N - \log S$ relationship corresponds to a Pareto distribution for the source fluxes, known to astrophysicists as a power-law model. Despite the unrealistic assumptions in the derivation, the linear $\log N - \log S$ relationship does have strong empirical support in a variety of contexts; e.g., Kenter and Murray (2003). In addition to its simplicity the power-law model also retains a high degree of interpretability, with the power-law exponent often of direct scientific interest. As a result of this simplicity and interpretability, the power-law model forms the basis of most $\log N - \log S$ analyses despite its many practical limitations in the ability to fit more complex datasets.

To address the limitations of this simple model astrophysicists have also experimented with a variety of broken power-law models. This is particularly important for larger populations or populations of sources spread over a wide energy range. Mateos *et al.* (2008) illustrate this by using both a two- and three-piece broken power-law model to capture the structure of the $\log N - \log S$ distribution across a wide range of energies. The basic idea of broken power-law models is to relax the assumption that the log survival function is a linear function of the log flux, and to instead assume a piecewise linear function. This adds additional challenges in estimating the location of the breakpoint, and quantifying the need for the breakpoint model above the simpler single power-law model. While recognizing the need to have more flexible models for $\log N - \log S$ analyses, most of the work in this area does not provide a coherent means to selecting the location and number of breakpoints.

Similarly to the single power-law model, the broken power-law model can be derived from first principles as a mixture of truncated and untruncated Pareto distri-

butions. The direct physical plausibility of the model is not as complete as for the single power-law model but the model parameters, in particular the slopes of the $\log N - \log S$ relationship can be used to draw conclusions about competing theories. The broken power-law provides a useful approximation that can be used to model mixtures of populations of sources, as well as more general piecewise-linear populations. Indeed, the broken power-law has empirical support in a variety of contexts both in astrophysics (Kouzu *et al.*, 2013; Mateos *et al.*, 2008) and outside (Segura *et al.*, 2013).

There are many alternative generalizations of the single power-law in addition to the broken power-law considered in this paper. For example, Ryde (1999) considers a smoothly broken power-law model that avoids the non-differentiability introduced by the strict broken power-law model. Other alternatives include mixtures of log-normal distributions and power-laws with modified tail behavior. In addition to parametric methods, the flux distribution can also be modeled nonparametrically. For the types of applications we are considering here, the main goal is parameter estimation and model selection to distinguish between single and broken power-law models. The scientific interpretability of a nonparametric model for the $\log(N > S) - \log(S)$ relationship is more complicated than the parametric alternative, and such approaches have gained less traction in the astrophysics community in the context of $\log N - \log S$ analyses. Therefore, while a more flexible nonparametric fit is perhaps statistically preferable, it is not as amenable to downstream science as in other contexts where the goal is prediction rather than estimation.

Among all generalizations, the strict broken power-law remains the most popular alternative. This popularity is a result of the interpretability of the model and the ease of translation from statistical results to scientific interpretability. Despite the popularity of the broken power-law model in the $\log N - \log S$ literature, there is currently no widely applicable and statistically rigorous method framework for fitting broken power-law models to the $\log N - \log S$ relationship to astrophysical source populations.

In this paper we provide an automatic method for jointly inferring the number and location of breakpoints and the parameters of interest for the $\log N - \log S$ problem. Our method allows astrophysicists to reliably infer both the number and the location of breakpoints in the $\log N - \log S$ relationship in a statistically rigorous manner for the first time. This simultaneous fitting introduces new computational challenges, so our method utilizes a new extension of the EM algorithm, known as the Interwoven EM Algorithm (IEM) (Baines, 2010; Baines *et al.*, 2012b). The IEM algorithm provides efficient and stable estimation of the model parameters across a wide range of parameter settings for a fixed number of breakpoints. To determine the number of breakpoints we then use an additional model selection procedure that employs the power posterior technique of Friel and Pettitt (2008) to accurately compute the log-likelihood of the candidate models.

The remainder of the paper is organized as follows. In Section 5.2 we introduce the necessary background and statistical formulation of the $\log N - \log S$ model. Section 5.3 provides details of our estimation procedure for a fixed number of breakpoints, with Section 5.4 outlining our model selection procedure to determine the number of breakpoints required. The performance of our method in terms of both parameter estimation and identification of the number of breakpoints is detailed in Section 5.5. An application to data from the *Chandra* Deep-Field North X-ray survey is provided in Section 5.6. Large-sample theory is developed in Section 5.7 and concluding remarks are offered in Section 5.8. Lastly technical details are given in an online supplement (Wong *et al.*, 2014) (see Appendix C).

5.2 Background and Problem Specification

Let $\boldsymbol{S} = (S_1, \ldots, S_n)^T$ denote a vector of the fluxes (in units of *ergs* s^{-1} cm^{-2}) of each of a population of n astrophysical sources. For example, we may be interested in the flux distribution of a selection of n X-ray pulsars located in a specified region of sky at a specified distance. The basic building block of our method is the power-law model:

$$N(> S) = \sum_{i=1}^{n} I_{\{S_i > S\}} \simeq \alpha S^{-\beta}, \quad S > \tau. \tag{5.1}$$

This specifies that the unnormalized survival function $N(> S)$ is approximately a power of the flux S. The power-law exponent, β, is the parameter of primary interest and provides domain specific knowledge about the source populations. The lower threshold τ can either be fixed according to the desired sensitivity level, or estimated from the data. Equivalently, taking the logarithm of both sides, (5.1) assumes a linear relationship between $\log(N(> S))$ and $\log(S)$:

$$\log(N(> S)) \simeq \log(\alpha) - \beta \log(S), \quad S > \tau. \tag{5.2}$$

In a statistical context, the theoretical power-law assumption corresponds to assuming that the source fluxes follow a Pareto distribution:

$$S_i \overset{\text{iid}}{\sim} \text{Pareto}(\beta, \tau), \quad i = 1, \ldots, n.$$

In practice, the linear $\log N - \log S$, or Pareto, assumption is not sufficient to describe the $\log N - \log S$ relationship for many real datasets. There are several ways to generalize (5.1), the most popular among astrophysicists being the broken power-law model as illustrated in Jordán *et al.* (2004) and Cappelluti *et al.* (2007). The starting point of the broken power-law is to replace (5.1) with a monotonically decreasing piecewise linear approximation. In the case of a two-piece model we assume:

$$\log(N(> S)) = \begin{cases} \log(\alpha_1) - \beta_1 \log(S), & \tau_1 < S \leq \tau_2, \\ \log(\alpha_2) - \beta_2 \log(S), & S > \tau_2, \end{cases} \tag{5.3}$$

where β_1 and β_2 are parameters of interest. Note that as a result of the continuity and normalization constraints on $\tau_1, \tau_2, \alpha_1, \alpha_2, \beta_1$ and β_2 there are a total of 4 free parameters in this expanded two-piece model. Applications of the broken power-law model in the astrophysics community typically use either fixed numbers and locations of the breakpoint(s) or selection via ad hoc procedures (Trudolyubov *et al.*, 2002). The contribution of this paper is the proposal of an automatic procedure for selecting the number and estimating the locations of the breakpoints jointly with the parameters of interest.

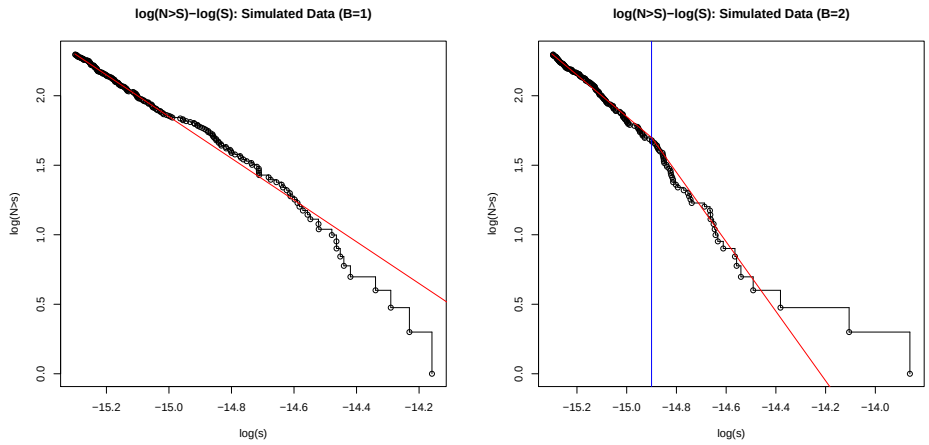


Figure 5.1. Example simulations of flux distributions under the single power-law model (left), and the broken power-law model (right). In practice, the fluxes are not directly observed and must be inferred from count data as described in Section 5.2. For the broken power-law example, the vertical blue line corresponds to the location of the breakpoint.

Figure 5.1 depicts the $\log N - \log S$ relationship for flux distributions simulated under a single power-law (left) and broken power-law model (right). As may be expected, even under a theoreticaly linear relationship, the empirical $\log N - \log S$-curve regularly exhibits non-linear features in the $\log N - \log S$-space. Depending on the difference in the power-law slopes, the breakpoint may be clearly visible, or indistinguishable by eye. In either case, it should be noted that much larger variations in the $\log N - \log S$ relationship are to be expected in the lower-right part of the curves as a result of the $\log - \log$ scaling. As will be seen in Section 5.2.1, the task of estimating the parameters controlling the flux distribution and/or detecting a breakpoint is additionally challenging because the fluxes depicted in Figure 5.1 are not directly observed.

5.2.1 Hierarchical modeling of the $\log N - \log S$ relationship

We now describe the connection between the broken power-law model introduced in (5.3) and the observed data. In practice the flux of each source, S_i, is not observed directly. Instead, we observe a Poisson-distributed photon count whose intensity is a known function of the parameter S_i. Let $Y_1, Y_2, \ldots, Y_n$ denote the source counts, then

we assume the following hierarchical model. For $i = 1, \ldots, n$,

$$Y_i | S_1, \ldots, S_n \overset{\text{indep.}}{\sim} \text{Poisson}(A_i S_i + b_i) \quad \text{and}$$
$$S_i \overset{\text{iid}}{\sim} \text{Pareto}_B(\boldsymbol{\beta}, \boldsymbol{\tau}), \tag{5.4}$$

where A_i's and b_i's are known constants (see below), $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_B) > \mathbf{0}$, $\boldsymbol{\tau} = (\tau_1, \ldots, \tau_B)$ such that $\tau_B > \cdots > \tau_1 > 0$, and $\text{Pareto}_B(\boldsymbol{\beta}, \boldsymbol{\tau})$ represents a B-piece Pareto distribution with survival distribution

$$S_B(x) = \begin{cases} 1, & x < \tau_1 \\ \left(\frac{\tau_1}{x}\right)^{\beta_1}, & \tau_1 \le x < \tau_2 \\ \left(\frac{\tau_1}{\tau_2}\right)^{\beta_1} \left(\frac{\tau_2}{x}\right)^{\beta_2}, & \tau_2 \le x < \tau_3 \\ & \vdots \\ \left\{\prod_{j=1}^{B-1} \left(\frac{\tau_j}{\tau_{j+1}}\right)^{\beta_j}\right\} \left(\frac{\tau_B}{x}\right)^{\beta_B}, & x \ge \tau_B \end{cases}$$

and thus its distribution function $F_B(\cdot) = 1 - S_B(\cdot)$. Note that the B-piece Pareto distribution corresponds to the broken power-law. The probability density f_B can be easily found by differentiation. When $B = 1$, the B-Pareto distribution reduces to a Pareto distribution with probability density function

$$f_1(x; \beta, \tau) = \begin{cases} \frac{\beta \tau^\beta}{x^{\beta+1}}, & x \ge \tau. \\ 0, & x < \tau. \end{cases}$$

In the above A_i's, sometimes known as effective areas, represent sensitivities of the detector, while b_i's represent background intensities. With the above model the goal is then to estimate B and, at the same time, $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$. At first sight, this seems to be a straightforward statistical problem: for a fixed B maximum likelihood estimation can be used to estimate $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$, while the issue of choosing B can be viewed as a model selection problem and thus traditional ideas such as AIC and BIC can be used. However, as to be seen below, practical implementation of these ideas poses serious computational challenges that cannot be easily solved.

5.3 Maximum Likelihood Estimation When B Is Known

In this section we provide details of how to obtain maximum likelihood estimates of $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$ for a fixed number of breakpoints B in the $\log N - \log S$ model. Defining $\beta_0 = 0$, $\tau_0 = \tau_1$ and $\tau_{B+1} = \infty$, the likelihood is

$$L(\boldsymbol{\beta}, \boldsymbol{\tau}; Y_1, \ldots, Y_n) = \prod_{i=1}^{n} \left\{ \int_{\tau_1}^{\infty} \frac{e^{-(A_i s + b_i)} (A_i s + b_i)^{Y_i}}{Y_i!} f_B(s; \boldsymbol{\beta}, \boldsymbol{\tau}) ds \right\}.$$

Note that the likelihood involves some numerically unstable integrals that do not have a closed form solution, and hence a direct maximization is extremely difficult. To further appreciate this difficulty, consider the case when there is no background contamination ($b_i = 0$), for which the above likelihood degenerates to

$$\prod_{i=1}^{n} \left[\sum_{j=1}^{B} \left(\frac{\tau_{j-1}}{\tau_j} \right)^{\beta_{j-1}} \frac{\beta_j (A_i \tau_j)^{\beta_j}}{Y_i!} \left\{ \Gamma(Y_i - \beta_j, A_i \tau_j) - \Gamma(Y_i - \beta_j, A_i \tau_{j+1}) \right\} \right].$$

Here, $\Gamma(a, x) = \int_x^{\infty} t^{a-1} e^{-t} dt$ is the incomplete gamma function which is numerically unstable, particularly when the first argument is large. Together with the inner summation in the above expression, these issues make a direct maximization of the (log-)likelihood difficult even when there is no background contamination. To address these issues we propose an EM-algorithm (Dempster *et al.*, 1977) to find the maximum likelihood estimators of $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$ for the general case of $b_i \geq 0$.

5.3.1 EM with a Sufficient Augmentation Scheme

The EM algorithm (Dempster *et al.*, 1977) has long been popular for its monotone convergence and resulting stability, and is therefore well-suited to our context. As always, the EM algorithm must be formulated in terms of "missing data" or auxiliary variables, that must be integrated out to obtain the observed data log-likelihood. For the current problem, since we are interested only in inference for $\boldsymbol{\beta}$ and $\boldsymbol{\tau}$, marginalizing over the uncertainty in the individual fluxes, it is natural to treat $\mathbf{S} = (S_1, \ldots, S_n)^T$ as the missing data. Since $\mathbf{S}$ is a sufficient statistic for $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{\tau})^T$, we call this the sufficient augmentation (SA) scheme in the terminology of Yu and Meng (2011).

Let $\mathbf{Y} = (Y_1, \ldots, Y_n)^T$. The complete data log-likelihood of $(\mathbf{Y}, \mathbf{S})$ is

$$\log p(\mathbf{Y}, \mathbf{S}; \boldsymbol{\beta}, \boldsymbol{\tau}) = \sum_{i=1}^{n} \log g(Y_i; A_i S_i + b_i) + \sum_{i=1}^{n} \log f_B(S_i; \boldsymbol{\beta}, \boldsymbol{\tau}),$$

where $g(x; \mu)$ is the probability mass function of a Poisson distribution with mean μ. In the E-Step of the algorithm we compute the conditional expectation

$$\begin{aligned} Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)}) &= \mathbb{E}\left\{\log p(\mathbf{Y}, \mathbf{S}; \boldsymbol{\theta}) | \mathbf{Y}; \boldsymbol{\theta}^{(k)}\right\} \\ &= \sum_{i=1}^{n} \mathbb{E}\left\{\log g(Y_i; A_i S_i + b_i) | Y_i; \boldsymbol{\theta}^{(k)}\right\} + \sum_{i=1}^{n} \mathbb{E}\left\{\log f_B(S_i; \boldsymbol{\theta}) | Y_i; \boldsymbol{\theta}^{(k)}\right\}, \quad (5.5) \end{aligned}$$

where $\boldsymbol{\theta}^{(k)}$ denotes the estimate of $\boldsymbol{\theta}$ at the k-th iteration. The M-step of the algorithm must then maximize $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ with respect to $\boldsymbol{\theta}$. Since the first term of (5.5) does not depend on $\boldsymbol{\theta}$, it can be ignored in our maximization. For the second term, as it does not admit a closed form expression, a Monte Carlo method is used to approximate it. The basic idea is to estimate it by the mean of a suitable Monte Carlo sample of the S_i's as described in Algorithm 1.

Without the first term in (5.5), the maximization of $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ is equivalent to finding the MLE of $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{\tau})^T$ from an iid sample $\boldsymbol{X} = (X_1, \ldots, X_m)$ from the Pareto$_B(\boldsymbol{\beta}, \boldsymbol{\tau})$ distribution. The log-likelihood of $\boldsymbol{X}$ is

$$\begin{aligned} l(\boldsymbol{\theta}; \boldsymbol{X}) &= \sum_{j=1}^{B} \beta_j \left(n_j \log \tau_j - n_{j+1} \log \tau_{j+1}\right) + \sum_{j=1}^{B} m_j \log \beta_j \\ &\quad - \sum_{j=1}^{B} \beta_j \sum_{i \in A_j} \log X_i - \sum_{i=1}^{m} \log X_i, \end{aligned}$$

where $n_j = \text{card}\{i\colon X_i \geq \tau_j\}$, $n_{B+1} = 0$, $m_j = n_{j+1} - n_j$, $\tau_{B+1} = \infty$, $n_{B+1} \log \tau_{B+1}$ is defined to be 0, and $A_j = \{i\colon \tau_j \leq X_i < \tau_{j+1}\}$. Note that the n_j's and m_j's are functions of $\boldsymbol{\tau}$. For any fixed $\boldsymbol{\tau}$, straightforward algebra shows that $l(\boldsymbol{\theta}; \boldsymbol{X})$ is maximized when β_j is set to

$$\beta_j(\boldsymbol{\tau}) = m_j(\boldsymbol{\tau}) \left(\sum_{i \in A_j} \log X_i + n_{j+1}(\boldsymbol{\tau}) \log \tau_{j+1} - n_j(\boldsymbol{\tau}) \log \tau_j \right)^{-1}, \quad (5.6)$$

$j = 1, \ldots, B$. By substituting the above expression, $l(\boldsymbol{\theta}; \boldsymbol{X})$ becomes

$$l(\boldsymbol{\theta}; \boldsymbol{X}) = -m - \sum_{i=1}^{m} \log X_i + \sum_{j=1}^{B} m_j(\boldsymbol{\tau}) \log \beta_j(\boldsymbol{\tau}). \tag{5.7}$$

Therefore, to obtain the MLE for $\boldsymbol{\theta} = (\boldsymbol{\beta}, \boldsymbol{\tau})^T$ from $\boldsymbol{X}$, one can first maximize $l(\boldsymbol{\theta}; \boldsymbol{X})$ in (5.7) with respect to $\boldsymbol{\tau}$, and then plug in the corresponding maximizer $\hat{\boldsymbol{\tau}}$ (i.e., the MLE of $\boldsymbol{\tau}$) into (5.6) to obtain the MLE $\hat{\boldsymbol{\beta}}$ for $\boldsymbol{\beta}$.

The MLE of τ_1 is $\hat{\tau}_1 = \min(X_1, \ldots, X_m)$, while unfortunately the MLEs for $\tau_2, \ldots, \tau_B$ do not admit closed-form expressions. Further, (5.7) is not a continuous function in $\boldsymbol{\tau}$ and therefore traditional optimization methods that require function derivatives (e.g., Newton-like methods) cannot be applied here. We have experimented with various optimization algorithms and found that the Nelder-Mead algorithm works well for this problem. The major steps of the EM algorithm in the SA scheme (SAEM) for finding the MLEs of $\boldsymbol{\theta}$ are given in Algorithm 1. In practice, the SAEM algorithm often converges very slowly. Section 5.3.4 below provides some illustrative numerical examples.

5.3.2 EM with an Ancillary Augmentation Scheme (AAEM)

Given the slow convergence of the SAEM algorithm, we seek faster alternatives. This subsection proposes an alternative EM algorithm that is based on an ancillary augmentation (AA) scheme, called the AAEM algorithm. For a discussion of augmentation schemes and their use in EM, see Baines *et al.* (2012b). The basis of our AAEM is to re-express our model using auxiliary variables $U_i = F_B(S_i; \boldsymbol{\theta})$:

$$\begin{aligned} Y_i | U_1, \ldots, U_n &\overset{\text{indep.}}{\sim} \text{Poisson}(A_i F_B^{-1}(U_i; \boldsymbol{\theta}) + b_i) \quad \text{and} \\ U_i &\overset{\text{iid}}{\sim} \text{Uniform}(0, 1), \end{aligned}$$

for $i = 1, \ldots, n$. Here $\mathbf{U} = (U_1, \ldots, U_n)$ is treated as the missing data, and preserves the observed data log-likelihood. In the E-Step we then calculate the conditional expectation

$$Q(\boldsymbol{\theta} | \boldsymbol{\theta}^{(k)}) = \sum_{i=1}^{n} \mathbb{E}\left\{ \log g(Y_i; A_i F_B^{-1}(U_i; \boldsymbol{\theta}) + b_i) | Y_i; \boldsymbol{\theta}^{(k)} \right\}. \tag{5.8}$$

Algorithm 1 SAEM: EM with the Sufficient Augmentation Scheme (SAEM)

1. Choose a starting value $\boldsymbol{\theta}^{(0)}$ and set $k = 0$.

2. Generate $\mathbf{S}^{(1)}, \ldots, \mathbf{S}^{(N_{\text{sim}})}$ from $p(\mathbf{S}|\mathbf{Y}; \boldsymbol{\theta}^{(k)})$ using the following Metropolis-Hastings algorithm. For each simulation of $\mathbf{S}$, we sample the elements of $\mathbf{S}$ one at a time. Suppose $\mathbf{S} = (S_1, \ldots, S_n)$ is the current draw. Denote $\mathbf{S}^* = (S_1, \ldots, S_{j-1}, S_j^*, S_{j+1}, \ldots, S_n)$, where S_j^* is drawn from $\text{Pareto}_B(\boldsymbol{\beta}^{(k)}, \boldsymbol{\tau}^{(k)})$. We accept this $\mathbf{S}^*$ as new value with probability $a_j(\mathbf{S}, \mathbf{S}^*)$; otherwise, we retain $\mathbf{S}$. The acceptance probability is given by

$$a_j(\mathbf{S}, \mathbf{S}^*) = \min\left\{1, \frac{g(Y_j; A_j S_j^* + b_j)}{g(Y_j; A_j S_j + b_j)}\right\}.$$

3. Find the maximizer $\tilde{\boldsymbol{\theta}}$ of the Monte Carlo estimate of $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$. This is equivalent to computing

$$\tilde{\boldsymbol{\theta}} = \underset{\boldsymbol{\theta}}{\text{argmax}} \frac{1}{N_{\text{sim}} - N_{\text{burn}}} \sum_{s=N_{\text{burn}}+1}^{N_{\text{sim}}} \sum_{i=1}^{n} \log f_B(S_i^{(s)}; \boldsymbol{\theta}),$$

   where N_{burn} is the number of burn-in. As discussed above, $\tilde{\boldsymbol{\theta}}$ can be obtained by the following steps:

   (a) set $\tilde{\tau}_1 = \min\{S_i^{(s)} : i = 1, \ldots, n, s = N_{\text{burn}} + 1, \ldots, N_{\text{sim}}\}$,

   (b) obtain $\tilde{\tau}_2, \ldots, \tilde{\tau}_B$ as the maximizer of $\sum_{j=1}^{B} m_j(\boldsymbol{\tau}^*) \log \beta_j(\boldsymbol{\tau}^*)$, where $\boldsymbol{\tau}^* = (\tilde{\tau}_1, \tau_2, \ldots, \tau_B)$, using the Nelder-Mead algorithm, and

   (c) set $\tilde{\beta}_j = \beta_j(\tilde{\boldsymbol{\tau}})$ using (5.6), for $j = 1, \ldots, B$.

4. Set $\boldsymbol{\theta}^{(k+1)} = \tilde{\boldsymbol{\theta}}$.

5. Repeat Steps 2 to 4 until convergence.

This conditional expectation can be approximated and maximized in a similar manner as for the $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$ in the SAEM algorithm. The resulting AAEM algorithm is summarized in Algorithm 2. Section 5.3.4 provides some empirical comparisons between the AAEM and SAEM algorithms. As may be expected, there are some situations where the AAEM algorithm converges faster, while there are other situations where the SAEM algorithm converges faster.

Algorithm 2 AAEM: EM with Ancillary Augmentation Scheme

1. Choose a starting value $\boldsymbol{\theta}^{(0)}$ and set $k = 0$.

2. Generate $\mathbf{U}^{(1)}, \ldots, \mathbf{U}^{(N_{\text{sim}})}$ from $p(\mathbf{U}|\mathbf{Y}; \boldsymbol{\theta}^{(k)})$ using the Metropolis-Hastings algorithm. For each simulation of $\mathbf{U}$, we sample the element of $\mathbf{U}$ one by one. Let $\mathbf{U} = (U_1, \ldots, U_n)$ be the previous draw. If we denote $\mathbf{U}^* = (U_1, \ldots, U_{j-1}, U_j^*, U_{j+1}, \ldots, U_n)$, where U_j^* is drawn from Uniform$(0,1)$. We accept this $\mathbf{U}^*$ as new value with probability $b_j(\mathbf{U}, \mathbf{U}^*)$; otherwise, we retain $\mathbf{U}$. The acceptance probability is given by

$$b_j(\mathbf{U}, \mathbf{U}^*) = \min\left\{1, \frac{g(Y_j; A_j F_B^{-1}(U_j^*; \boldsymbol{\theta}^{(k)}) + b_j)}{g(Y_j; F_B^{-1}(U_j; \boldsymbol{\theta}^{(k)}) + b_j)}\right\}.$$

3. Find the maximizer $\tilde{\boldsymbol{\theta}}$ of the following Monte Carlo estimate of $Q(\boldsymbol{\theta}|\boldsymbol{\theta}^{(k)})$:

$$\frac{1}{N_{\text{sim}} - N_{\text{burn}}} \sum_{s=N_{\text{burn}}+1}^{N_{\text{sim}}} \sum_{i=1}^{n} \log g(Y_i; A_i F_B^{-1}(U_i^{(s)}; \boldsymbol{\theta}) + b_i).$$

   The maximization can be done for example with the Nelder-Mead algorithm.

4. Set $\boldsymbol{\theta}^{(k+1)} = \tilde{\boldsymbol{\theta}}$.

5. Repeat Steps 2 to 4 until convergence.

5.3.3 Interwoven EM (IEM)

In practice, choosing the most efficient algorithm between the SAEM and AAEM requires knowledge of the unknown parameter values and the theoretical convergence

rates, both of which are not available in most contexts. Therefore, it would instead be desirable if one could combine the "best parts" of SAEM and AAEM rather than select one of them. One simple way to combine the two algorithms is to use the so-called alternating EM (AEM) algorithm. The AEM algorithm proceeds by using SAEM for the first iteration, then uses AAEM for the second iteration, followed by SAEM for the third, and so on. While this procedure tends to "average" the performance of the two algorithms, a more sophisticated way to combine them is to use the Interwoven EM (IEM) algorithm of Baines *et al.* (2012b). Theoretical and empirical results show that IEM typically achieves sizeable performance gains over the component EM algorithms. The key to the boosted performance of IEM is that it utilizes the joint structure of the two augmentation schemes through a special "IE-Step". In contrast, AEM simply performs sequential updates using each augmentation scheme that make no use of this joint information. The theory of the IEM algorithm in Baines *et al.* (2012b) shows that the rate of convergence of IEM is dependent on the "correlation" between the two component augmentation schemes. Since the SA and AA schemes typically have low correlation, here we interweave these two schemes to produce an IEM algorithm for estimating the parameters of flux distributions.

The IEM algorithm for our $\log N - \log S$ model is given in Algorithm 3. The algorithm requires very minimal computation in addition to the component SAEM and AAEM algorithms so is comparable in real-time per-iteration speed. Lastly we note that there is some freedom in how to combine the IEM algorithm with MC methods. Specifically, there are variations in how one may choose to implement Step 3. One may want to sample U again instead of using the previous samples in Step 2. In both cases, one obtains a sample from $\mathbf{U}|\boldsymbol{Y}, \boldsymbol{\theta}^{(k+0.5)}$ and achieves the goal. From our practical experience, we found that there is very little difference between the performances of these two approaches. Thus, we choose to use the one which is least computationally expensive.

Algorithm 3 IEM: Interwoven EM

1. Choose a starting value $\boldsymbol{\theta}^{(0)}$ and set $k = 0$.

2. Execute Steps 2 and 3 of the SAEM algorithm. Set $\boldsymbol{\theta}^{(k+0.5)} = \tilde{\boldsymbol{\theta}}$.

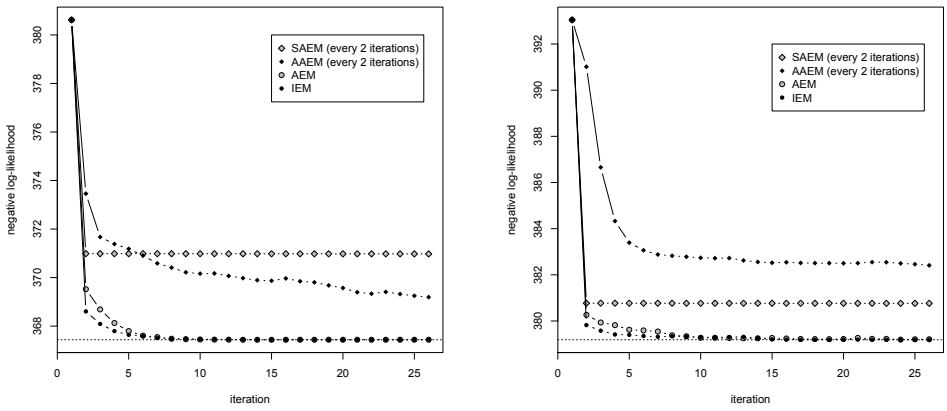
3. Execute Step 3 of the AAEM algorithm, with $\mathbf{U}^{(l)}$ generated as: $U_j^{(l)} = F_B(S_j^{(l)}; \boldsymbol{\theta}^{(k+0.5)})$, for $j = 1, \ldots, n$ and $l = N_{\text{burn}} + 1, \ldots, N_{\text{sim}}$. Set $\boldsymbol{\theta}^{(k+1)} = \tilde{\boldsymbol{\theta}}$.

4. If convergence is achieved or k attains N_{limit}, then declare $\boldsymbol{\theta}^{(k+1)}$ to be MLE; otherwise set $k = k + 1$ and return to Step 2.

5.3.4 An Empirical Comparison Amongst Different EM Algorithms

In this subsection we empirically compare the convergence speeds of SAEM, AAEM, AEM and IEM by applying them to two simulated data sets. These two data sets were simulated from a model with $B = 1$ and no background contamination counts. This model is somewhat simple but the advantage is that the likelihood function simplifies considerably, and the corresponding maximum likelihood estimates can be reliably obtained with non-EM methods. With these maximum likelihood estimates the maximized log-likelihood value can be calculated and used for baseline comparisons.

In Figure 5.2(a), for the first simulated data set, we plot the negative log-likelihood values of the SAEM, AAEM, AEM and IEM estimates evaluated at different iterations. One can see the slow convergence speeds of SAEM and AAEM, with SAEM being the slower. Also, both AEM and IEM converged relatively fast, with IEM being the faster. When comparing to AEM, IEM utilizes the relationship between SAEM and AAEM at each step, which leads to the superiority of IEM. As noted earlier, the convergence rate of IEM is heavily influenced by the 'correlation' between the two data augmentation schemes being interwoven; i.e., the SA and AA for this example. For the $\log N - \log S$ model the correlation between these augmentation schemes is hard to estimate exactly, but it appears empricially that the SA and AA have a reasonably high correlation, thus preventing IEM from outperforming AEM by a larger amount. This is likely due to



(a) Simulated Data Set 1 (b) Simulated Data Set 2

Figure 5.2. Plots of negative log-likelihood values for different EM algorithms. In each plot the horizontal dashed line indicates the negative log-likelihood evaluated at the maximum likelihood estimates.

τ, which controls the boundary of the parameter of the space and heavily impacts the rate of convergence. However, among the candidate algorithms IEM yields the best convergence properties.

We repeat the same plot in Figure 5.2(b) for the second simulated data set. This time the relative speeds of SAEM and AAEM switched; i.e., SAEM converged faster. This illustrates that neither SAEM or AAEM is uniformly superior to the other across all datasets. The relative rate of convergence of AEM and IEM remain the same for these two datasets and across other simulated datasets (not shown).

Overall from these two plots one can see that the IEM algorithm is the most efficient and robust. Also, when comparing to AEM, it is computationally faster due to the skipping of an extra sampling step. Similar performance was observed across a wide range of simulation settings. Therefore we recommend using the IEM algorithm to compute the maximum likelihood estimates when B is known.

5.4 Automated Choice of B

This section addresses the important problem of selecting the number of "pieces", B, in the broken-Pareto model. Since this problem can be seen as a model selection

problem, one can adopt well studied methods such as AIC and BIC to solve it. To proceed we first note that when $B = 1$, the number of free parameters in the model is $2B$. With AIC, the best B is chosen as

$$\hat{B}_{\text{AIC}} = \underset{B}{\text{argmax}}\, \text{AIC}(B) = \underset{B}{\text{argmax}} \left\{ -2 \log L(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\tau}}; Y_1, \ldots, Y_n) + 4B \right\},$$

while for BIC B is chosen as the minimizer of

$$\hat{B}_{\text{BIC}} = \underset{B}{\text{argmax}}\, \text{BIC}(B) = \underset{B}{\text{argmax}} \left\{ -2 \log L(\hat{\boldsymbol{\beta}}, \hat{\boldsymbol{\tau}}; Y_1, \ldots, Y_n) + 2B \log n \right\}.$$

Despite the straightforward definitions, in practice the numerical instability of the likelihood function makes computation of $\text{AIC}(B)$ and $\text{BIC}(B)$ very challenging. To address this problem we adopt the so-called power posterior method proposed by Friel and Pettitt (2008) to approximate the log-likelihood directly.

In our context, the power posterior is defined as

$$p_t(\mathbf{S}|\mathbf{Y}; \boldsymbol{\theta}) \propto p(\mathbf{Y}|\mathbf{S})^t p(\mathbf{S}; \boldsymbol{\theta}) \quad \text{for } 0 \leq t \leq 1.$$

In addition, define

$$z(\mathbf{Y}|t) = \int_{\mathbb{R}^n} p(\mathbf{Y}|\mathbf{s})^t p(\mathbf{s}; \boldsymbol{\theta}) d\mathbf{s},$$

and, for simplicity, write the likelihood as $p(\mathbf{Y}) = L(\boldsymbol{\beta}, \boldsymbol{\tau}; Y_1, \ldots, Y_n)$. The following equality is crucial to this method:

$$\log\{p(\mathbf{Y})\} = \log \left\{ \frac{z(\mathbf{Y}|t=1)}{z(\mathbf{Y}|t=0)} \right\} = \int_0^1 \mathbb{E}\left[\log\{p(\mathbf{Y}|\mathbf{S})\} | \mathbf{Y}; \boldsymbol{\theta}, t \right] dt,$$

where the last expectation (inside the integral) is taken with respect to the power posterior $p_t(\mathbf{S}|\mathbf{Y}; \boldsymbol{\theta})$. The idea is as follows. First, for any given t, Monte Carlo methods can be applied to sample from the power posterior and approximate the expectation. Once a sufficient number of these expectations (corresponding to different values of t) are calculated, numerical methods can be used to approximate the integral, which is the same as the log-likelihood. Since this method approximates the log-likelihood directly (i.e., without the computation of the likelihood), it is numerically quite stable. The detailed algorithm is presented as Algorithm 4.

Algorithm 4 Power Posterior Method for Log-Likelihood Calculation

1. Choose a starting value $\mathbf{S}^{(0)}$ and set $k = 0$.

2. Set $t = (k/N_{\text{grid}})^c$, where c controls the density of the grid values of t. It is typically set to 3 or 5 (see Friel and Pettitt, 2008).

3. Generate $\mathbf{S}^{(1)}, \ldots, \mathbf{S}^{(N_{\text{sim}})}$ from $p_t(\mathbf{S}|\mathbf{Y};\boldsymbol{\theta})$ using the Metropolis-Hastings algorithm described in Step 2 of the SAEM algorithm. Note that the acceptance probability becomes

$$a_j(\mathbf{S}, \mathbf{S}^*) = \min\left\{1, \left\{\frac{g(Y_j; A_j S_j^* + b_j)}{g(Y_j; A_j S_j + b_j)}\right\}^t\right\}.$$

4. Estimate $\mathbb{E}\left[\log\{p(\mathbf{Y}|\mathbf{S})\}|\mathbf{Y};\boldsymbol{\theta}, t\right]$ with

$$\hat{l}_t = \frac{1}{N_{\text{sim}} - N_{\text{burn}}} \sum_{s=N_{\text{burn}}+1}^{N_{\text{sim}}} \log p(\mathbf{Y}|\mathbf{S}^{(s)};\boldsymbol{\theta}).$$

5. If $k < N_{\text{grid}}$, set $k = k + 1$, $\mathbf{S}^{(0)} = \sum_{s=N_{\text{burn}}+1}^{N_{\text{sim}}} \mathbf{S}^{(s)}/(N_{\text{sim}} - N_{\text{burn}})$, and go to Step 2. Otherwise go to the next step.

6. Given the $\hat{l}_t$'s, the log-likelihood $\log\{p(\mathbf{Y})\}$ can be approximated via any reliable numerical integration method.

The above algorithm provides a reliable method for approximating the log-likelihood for a given value of $\boldsymbol{\theta}$. Then one natural question to ask is, can we not simply obtain the MLE of $\boldsymbol{\theta}$ by directly maximizing this log-likelihood approximation via, say, Newton's method? The answer, in principle, is yes, but the IEM algorithm is still preferred mainly because the estimates from IEM are generally more stable and reliable. Moreover, the power posterior approximation to the log-likelihood is computationally intensive if one wants to obtain an accurate estimate. For these reasons, we only use this power posterior approximation to estimate the log-likelihood evaluated at the MLE obtained by the IEM algorithm.

5.5 Simulation Experiments

Numerical experiments were conducted to evaluate the practical performance of the proposed methodology. Four experimental settings were considered:

1. $B = 1$, $\boldsymbol{\tau} = 5 \times 10^{-17}$, $\boldsymbol{\beta} = 1$ and $n = 100$,
2. $B = 2$, $\boldsymbol{\tau} = (1 \times 10^{-17}, 5 \times 10^{-17})^T$, $\boldsymbol{\beta} = (0.5, 3)^T$ and $n = 200$,
3. $B = 2$, $\boldsymbol{\tau} = (1 \times 10^{-17}, 5 \times 10^{-17})^T$, $\boldsymbol{\beta} = (0.5, 1.5)^T$ and $n = 200$,
4. $B = 3$, $\boldsymbol{\tau} = (1 \times 10^{-17}, 8 \times 10^{-17}, 1.8 \times 10^{-16})^T$, $\boldsymbol{\beta} = (0.3, 1, 3)^T$ and $n = 500$.

The parameter values of these settings were chosen to mimic the typical behavior of the real data. The effective areas and the expected background counts are set to $A_i = 10^{19}$ and $b_i = 10$ respectively for all i.

Two hundred data sets were generated for each experimental setting. For each generated data set, both AIC and BIC were applied to choose the value of B, and model parameters were estimated by the IEM algorithm. The selected values of B are summarized in Table 5.1. One can see that BIC works substantially better than AIC for selecting B, and while BIC occasionally overestimates B, there is a clear tendency for AIC to consistently overestimate B.

Other crucial factors that determine the ability of our method to detect structural breaks in the population distribution include: (i) the sample size, (ii) the separation

between breakpoints, and, (iii) the magnitude of the difference between the power-law slopes on adjacent segments. The impact of the third factor can be seen by comparing simulation results from settings 2 and 3, where the misclassification rate is seen to increase as the slopes become closer. From additional simulations our experience suggests that in typical settings a sample size of 200 or more is needed to reliably detect a single breakpoint, with double this required to detect two breakpoints. In simulations, true breakpoints can be detected for smaller sample sizes, but at a lower rate that is more dependent on the noise properties of the specific simulation.

In addition to selecting the number of breakpoints, we also conducted a simulation to assess the quality of parameter estimation when using the IEM algorithm. For each experimental setting, we calculated the squared error $(\beta_1 - \hat{\beta}_1)^2$ of $\hat{\beta}_1$ for all those data sets where $\hat{B}$ were correctly selected. We then computed the average of all these squared errors, denoted as $\text{mse}(\hat{\beta}_1)$, and calculated the relative mean squared error $\sqrt{\text{mse}(\hat{\beta}_1)}/\beta_1$. Similar relative mean squared errors for other estimates in $\hat{\boldsymbol{\beta}}$ and $\hat{\boldsymbol{\tau}}$ were obtained in a similar manner. These relative mean squared errors are given in Table 5.2. We note that all of these are of the order of 10^{-2} or 10^{-1}.

Table 5.1. The number of pieces $\hat{B}$ selected by AIC and BIC.

Experimental Setting	Model Selection Method	$\hat{B}$ 1	2	3	4
1	AIC	94	53	35	18
	BIC	164	33	3	0
2	AIC	0	135	45	20
	BIC	0	198	2	0
3	AIC	0	110	71	19
	BIC	0	177	23	0
4	AIC	0	0	138	62
	BIC	0	0	194	6

Table 5.2. The relative mean squared errors of $\hat{\beta}$ and $\hat{\tau}$, conditional on selection of the correct B. All entries are multiplied by 10^2.

Setting	Method	$\hat{\tau}$			$\hat{\beta}$		
1	AIC	5.14	-	-	11.1	-	-
	BIC	4.91	-	-	10.6	-	-
2	AIC	3.33	2.55	-	9.81	11.3	-
	BIC	3.52	2.60	-	9.17	10.8	-
3	AIC	3.52	14.2	-	12.0	13.2	-
	BIC	3.57	12.9	-	11.1	13.5	-
4	AIC	2.71	3.26	5.04	7.08	9.91	12.3
	BIC	2.72	3.94	4.97	7.16	9.74	11.9

5.6 Application: *Chandra* Deep Field North X-Ray Data

We now apply our method to data from the *Chandra* Deep Field North (CDFN) X-ray survey. Our dataset comprises a total of 225 sources with an off-axis angle of 8 arcmins or less and counts ranging from 5 to 8655. The full CDFN dataset is comprised of multiple observations at many different aimpoints, however we here consider only a subset where the aimpoints are close to each other to avoid complications such as variations in detection probability due to changes in the point spread function (PSF) shape, and consequent variations in detection probability. The decision to include only aimpoints close to each other was taken primarily to avoid the issue of 'incompleteness' and essentially amounts to taking a higher signal to noise subset of the full dataset. Incompleteness occurs when sources are not observed, typically a result of being too faint to be detected under the specific detector configuration used. Since this missingness is a function of the quantity to be estimated, it must be accounted for, and can lead to tremendously more complicated and challenging modeling. This approach is taken as part of a fully Bayesian analysis in Baines *et al.* (2012a), but there are significant challenges to the method. Most notably, results are very sensitive to the

'incompleteness function', which is frequently not known to such high precision. By considering only a subset of aimpoints we focus on a higher SNR subset of the *Chandra* data that is not subject to issues arising from incompleteness. We do not believe that the subset choice impacts the final conclusion, as the results in the unpublished report of Udaltsova, which models the full dataset and accounts for incompleteness, are extremely similar to those presented here. Since the off-axis angle measures the radial distance of the source from the center of the detector, sources with large off-axis angles can be thought of as being "close to the edge of the image". Sources appearing at large off-axis angles appear much larger and at lower resolution than those closer to the center of the detector. The source-specific scaling constant, effective area A_i, is used to account for variations in the expected number of photons as a function of source location and photon energy. However, at large off-axis angles additional complications such as "confusion" (two or more sources overlapping and appearing as one) and "incompleteness" (possible non-detections of fainter sources) must be considered. For the purposes of our analysis here, we include all sources with an off-axis angle < 8 arcmin to achieve a worst-case completeness of 80%. We also consider thresholding at < 6 and < 7 arcmins, with a full discussion of the sensitivity to this threshold considered in Section 5.6.1.

Applying our model selection procedure to the dataset with < 8 arcmins yields an estimate of $\hat{B} = 2$, with $\hat{B} = 1$ for the < 6 and < 7 arcmin subsets. As discussed in detail in Section 5.6.1, the consistency of the observations in the $6 - 8$ arcmin range suggests that the ability to detect the presence of a breakpoint is limited by the small sample sizes at < 6 and < 7. Figure 5.3 shows the $\log N - \log S$ plot for the < 8 arcmin dataset, depicting the log (base 10) of the empirical survival count as a function of the log flux, using the imputed fluxes from the final E-step of our algorithm. While the plot ignores the uncertainty in the S_i's, it remains the standard plot for the analysis of $\log N - \log S$ relationships. We note from the plot that the "break" is clearly visible around $\log_{10}(\tau_1) = -15.657$, with a change in slope from 0.48 to 0.85. Full parameter estimates and standard error estimates are provided in Table 5.3. Standard error

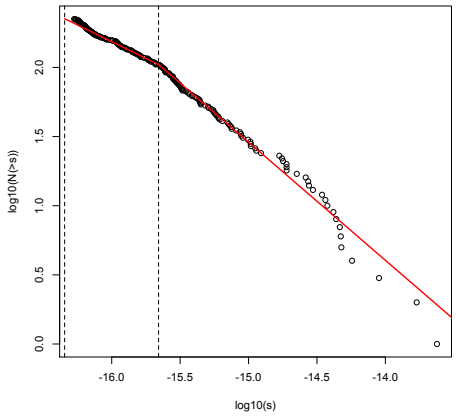


Figure 5.3. $\log N - \log S$ plot for the *Chandra* Deep Field North data with off-axis angle truncation at 8 arcmins. The vertical dotted lines are drawn at $\hat{\tau}_1$ and $\hat{\tau}_2$. The red lines correspond to the fitted broken-Pareto model with estimated slopes $\hat{\beta}_1$ and $\hat{\beta}_2$.

estimates are obtained using a simple Bootstrap resampling procedure. We also note that by simulating from the model, the seemingly nonlinear behavior of the curve at $\log(S) = -14.5$ is nonetheless seen to be consistent with the piecewise linear model. Our analysis shows that a two-piece broken power-law model is preferred for this subset, with a breakpoint at a lower flux than shown in Moretti *et al.* (2003), and with the lower segment at a flatter slope. This differs from what would be expected if point sources are to make up all of the diffuse background (Hickox and Markevitch, 2007), suggesting that a significant proportion of the residual X-ray background is composed of diffuse emission (e.g., hot intergalactic plasma); see also Mateos *et al.* (2008).

The analysis in Hickox and Markevitch (2007) was based on optical sources from the Hubble Space Telecope (HST) which had no X-ray counterparts. By considering various models for the X-ray intensities of these sources, Hickox and Markevitch (2007) compared them to the residual X-ray background from deep Chandra observations. The proportion of the Cosmic X-ray Background (CXB) that can be explained by point sources alone is typically around 70-80%. Connecting to our results, higher values for β_1 increase the possibility that deeper observations could be obtained that would explain an additional proportion of the CXB as discrete sources. Alternatively,

lower values for β_1 signify a flatter $\log N - \log S$, suggesting a greater amount of diffuse emission. Figure 8 of Hickox and Markevitch (2007) depicts the relationship between the proportion of the 0.5-2 keV CXB from unresolved HST point sources and the power law slope. The breakpoint estimated in our analysis translates to $\approx 10^{-16}$ergs^{-1}cm^{-2} for the passbands used by Hickox and Markevitch (2007). However, in the 2 Msec dataset they analyze, they do not detect any breakpoints (see their Figure 7). Our analysis indicates that the $\log N - \log S$ curve flattens for fluxes less than the breakpoint, thus allowing for a significant proportion of the unresolved residual X-ray background to be due to diffuse emission.

Table 5.3. Parameter estimates and standard errors for the CDFN dataset.

Parameter	Estimate	SE
β_1	0.483	0.060
β_2	0.854	0.224
$\log_{10}(\tau_1)$	-16.344	0.030
$\log_{10}(\tau_2)$	-15.657	0.271

5.6.1 CDFN Source Selection

In this section we consider the sensitivity of our analysis to the chosen off-axis angle threshold. As discussed in Section 5.6, at higher off-axis angles there are additional complications such as incompleteness and confusion that must be built into any statistical analysis that are not covered by the method presented here. Let K denote the maximum off-axis angle; i.e., all sources with off-axis angle less than K are retained, all others are excluded from the analysis. The choice of $K = 8$ for our analysis in Section 5.6 is motivated by scientific considerations and an estimated completeness above 80% at $K = 8$. However, by varying the truncation point we obtain additional insight into the sensitivity of our analysis to this decision, as well as to the statistical senstitivity to the sample size required for breakpoint detection. Table 5.4 shows the results of the analysis for differing values of K. As explained, results for $K > 9$ are

likely to be untrustworthy, although they happen to be similar to those with $K = 8$. On the other extreme, if we truncate at $K = 4$ or $K = 5$ we unnecessarily discard a large number of sources.

We note that at $K = 7$ we are also no longer able to formally detect a break i.e., $\hat{B} = 1$. However, upon closer examination the BIC values for $B = 1$ and $B = 2$ when $K = 7$ are very similar (2186.79 vs. 2188.37), indicating that there is little to choose between the $B = 1$ and $B = 2$ models. With a few additional data points added at $K = 8$, our procedure then has enough power to detect the break at $K = 8$. It is worth noting that all additional data points with off-axis angle between 7 and 8 were manually screened, and are quantitatively very similar to those with $K < 7$. That is, the detection (or lack) of a breakpoint in this context appears to be primarily determined by the sample size of the dataset used. This is consistent with our results from the simulation study in Section 5.5, where a sample size of approximately 200 was required to reliably detect a break with similar parameter configurations. Indeed, looking at the plot in Figure 5.3, we note that the break is rather a subtle one, with the estimated slopes differing by approximately 0.37. In summary, for this particular dataset we note that there appears to be evidence of a breakpoint, although the sample size required to detect the breakpoint is not reached until we truncate at $K = 8$, just before additional modeling considerations such as incompleteness must be accounted for.

5.7 Theoretical Properties

This section deals with the large-sample properties of the proposed procedure. We first establish consistency results for the case when B is known, with no background contamination ($b_i = 0$ for all i) and all A_i are assumed to be identical. Then we describe how one could weaken the assumptions of identical A_i's and zero b_i's. However, as explained at the end of this section, the case of unknown B is substantially more difficult and we are unable to provide any theoretical results for this case.

If it is assumed that $A_i = A > 0$ and $b_i = 0$ for all $i = 1, \ldots, n$, then $Y_1, \ldots, Y_n$

Table 5.4. CDFN Results by varying off-axis truncation

		$\log_{10}(\hat{\tau})$		$\hat{\beta}$	
K	n	$\log_{10}(\hat{\tau}_1)$	$\log_{10}(\hat{\tau}_2)$	$\hat{\beta}_1$	$\hat{\beta}_2$
4	77	−16.364		0.788	
5	112	−16.353		0.738	
6	152	−16.329		0.691	
7	192	−16.373		0.590	
8	225	−16.343	−15.668	0.482	0.850
9	257	−16.352	−15.732	0.449	0.850
10	287	−16.378	−15.696	0.450	0.792
11	298	−16.389	−15.702	0.456	0.793
12	303	−16.403	−15.677	0.454	0.802
13	304	−16.429	−15.843	0.412	0.743

constitute an iid sample from model (5.4). Denote the density of Y_1 by

$$\begin{aligned} f(y;\boldsymbol{\theta}) &= \int_{\tau_1}^{\infty} \frac{e^{-As}(As)^y}{y!} f_B(s;\boldsymbol{\beta},\boldsymbol{\tau})ds \\ &= \sum_{j=1}^{B} \left(\frac{\tau_{j-1}}{\tau_j}\right)^{\beta_{j-1}} \frac{\beta_j (A\tau_j)^{\beta_j}}{y!} \left\{\Gamma(y-\beta_j, A\tau_j) - \Gamma(y-\beta_j, A\tau_{j+1})\right\}. \end{aligned}$$

The parameter space is defined as $\Theta = \{\boldsymbol{\theta} = (\boldsymbol{\beta},\boldsymbol{\tau})^T \in \mathbb{R}_+^{2B} \colon \beta_j \neq \beta_{j+1}, \tau_j < \tau_{j+1}, j = 1,\ldots,B-1\}$. Let $\boldsymbol{\theta}_0 = (\boldsymbol{\beta}_0,\boldsymbol{\tau}_0)^T \in \Theta$ denote the true parameter value. Notice that Θ is not compact and that the value of the likelihood does not converge to zero if the parameter approaches the boundary of Θ. Therefore standard arguments such as the ones based on Wald (1949) do not apply directly in order to establish strong consistency of the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$ of $\boldsymbol{\theta}_0$. Instead a compactification device is applied to subsequently use the results of Kiefer and Wolfowitz (1956). This leads to the following result.

Theorem 5.1. *Suppose B is known and $A_i = A > 0$ for all $i = 1,\ldots,n$. Then, the maximum likelihood estimator $\hat{\boldsymbol{\theta}}$ is strongly consistent for $\boldsymbol{\theta}_0$, that is, $\hat{\boldsymbol{\theta}} \to \boldsymbol{\theta}_0$ with probability one as*

$n \to \infty$.

The proof of Theorem 5.1 is provided in an online supplement (Wong *et al.*, 2014) (see Appendix C). To weaken the restriction of identical A_i, observe that this condition is mainly applied to allow the use of the strong law of large numbers for iid random variables, as required for the direct application of the results in Wald (1949) and Kiefer and Wolfowitz (1956). Since the arguments used to prove Theorem 5.1 are still valid if only the assumption $A_i > 0$ is made, Kolmogorov's version of the strong law of large numbers can be applied to adapt their proof to the present case, imposing additional assumptions such as the Kolmogorov criterion

$$\sum_{i=1}^{\infty} \frac{\mathrm{Var}(Y_i)}{i^2} < \infty$$

or conditions ensuring the validity of Kolmogorov's three-series theorem. Then, the result of Theorem 5.1 holds also in this more general setting. The case for non-zero b_i's can also dealt with similarly, but with long and tedious algebra.

In the theory developed above, the number of pieces, B, in the broken-Pareto model is assumed to be known. The case of unknown B is, however, substantially more difficult. In fact, results in simpler settings such as the traditional "change in mean" scenario, in which segments of independent observations differ only by their levels, strong distributional assumptions become necessary to show consistency of an estimator for B. These typically require normality of the observations so that sharp tail estimates of the supremum of certain Gaussian processes are available; e.g., see Yao (1988). These techniques have also been exploited in Aue and Lee (2011) for image segmentation purposes. However, in the current context of the more complex broken-Pareto model, these arguments are not applicable and in fact it seems infeasible to derive theoretical properties under a set of practically relevant assumptions.

5.8 Concluding Remarks

We provide a coherent statistical procedure for selecting the number and orientation of "pieces" in an assumed piecewise linear $\log N - \log S$ relationship. Our framework

allows astrophysicists to use a principled approach to reliably select the model order B, and for parameter estimation via maximum likelihood estimation in a numerically challenging context. To our knowledge, this is the first statistically rigorous procedure developed for solving this important scientific problem. *R* code implementing the proposed procedure can be obtained from the authors.

Appendix A

Supplement to "Fiber Direction Estimation in Diffusion MRI"

A.1 Estimation of the linear model (2.6)

This section describes a fast algorithm that we developed for estimating $\hat{\beta}_j$ in model (2.6). With this model one can write the log-likelihood of $\boldsymbol{\beta} = (\beta_1, \ldots, \beta_K)^\intercal$ as

$$\ell(\boldsymbol{\beta}) = \sum_{i=1}^{m} \left[\log\left(\frac{y_i}{\sigma^2}\right) - \frac{y_i^2 + (\sum_{k=1}^{K} \beta_k x_{ik})^2}{2\sigma^2} + \log I_0 \left\{ \frac{y_i(\sum_{k=1}^{K} \beta_k x_{ik})}{\sigma^2} \right\} \right],$$

where $y_i = S(u_i)$ and $x_{ik} = x_k(u_i)$ for $i = 1, \ldots, m, k = 1, \ldots, K$. And now we consider minimizing

$$-\ell(\boldsymbol{\beta}) \qquad \text{subject to} \qquad \beta_k \geq 0 \quad \forall k \tag{A.1}$$

with respect to $\boldsymbol{\beta}$. Now, differentiating ℓ with respect to β_j, we have

$$\frac{\partial \ell}{\partial \beta_j} = \sum_{i=1}^{m} \left\{ -\frac{(\sum_{k=1}^{K} \beta_k x_{ik}) x_{ij}}{\sigma^2} + \frac{y_i x_{ij}}{\sigma^2} t_i(\boldsymbol{\beta}) \right\},$$

where

$$t_i(\boldsymbol{\beta}) = I_1 \left\{ \frac{y_i(\sum_{k=1}^{K} \beta_k x_{ik})}{\sigma^2} \right\} \Big/ I_0 \left\{ \frac{y_i(\sum_{k=1}^{K} \beta_k x_{ik})}{\sigma^2} \right\}$$

with

$$I_v(x) = \frac{1}{\pi} \int_0^{\pi} \exp(x \cos\phi) \cos(v\phi) d\phi$$

as the v-th (for nonnegative integer v) order modified Bessel function of the first kind (Abramowitz and Stegun, 1964). One can show that the solution $\hat{\boldsymbol{\beta}}$ of minimizing (A.1) satisfies

$$\hat{\boldsymbol{\beta}}_j = \left[\frac{\sum_{i=1}^{m} \left\{ t_i(\hat{\boldsymbol{\beta}}) y_i - \sum_{k \neq j} \hat{\beta}_k x_{ik} \right\} x_{ij}}{\sum_{i=1}^{m} x_{ij}^2} \right]_+ \qquad \forall j. \tag{A.2}$$

If we know $t_i(\hat{\boldsymbol{\beta}})$'s, (A.2) gives an update formula for one β_k at a time, similarly as in common coordinate descent algorithms. Since coordinate descent algorithm is of an iterative basis, we propose to further approximate $t_i(\hat{\boldsymbol{\beta}})$ by substituting the latest update of $\boldsymbol{\beta}$ into t_i. This leads to the following coordinate descent like strategy for finding $\hat{\boldsymbol{\beta}}$:

- Outer loop: Approximate $r(\hat{\boldsymbol{\beta}})$ using the latest update of $\boldsymbol{\beta}$.
- Inner loop: Coordinate updates through (A.2) until convergence.

For inner loop, very often, many coefficients remain zero after thresholding, which leads to unchanged of their values. Since the update of a particular coefficient depends on the partial sum of other coefficients, the inner loop is usually computationally efficient and converges in a fast manner.

This algorithm requires an initial value of $\boldsymbol{\beta}$. Motivated by the typical non-linear estimator of a single fiber model, we can choose the initial value as a constrained least square estimator which minimizes

$$\sum_{i=1}^{m} \left(y_i - \sum_{k=1}^{K} \beta_k x_{ik} \right)^2 \qquad \text{subject to} \qquad \beta_k \geq 0 \qquad \forall k.$$

Note that this is a quadratic programming problem, which can be solved efficiently by existing algorithms.

A.2 Simulation study of voxel-wise estimation

This section provides simulation results for the voxel-wise estimation procedure proposed in Section 2.3. Observed signal intensities were simulated from model (2.2) with Rician noise under three settings:

1. Single tensor case: $J = 1$, $\mathbf{m}_1 = (1, 0, 0)^\intercal$.

2. Two tensor case with perpendicular crossing and unbalanced components: $J = 2$, $\mathbf{m}_1 = (1, 0, 0)^\intercal$, $\mathbf{m}_2 = (0, 1, 0)^\intercal$, $p_1 = 0.7$, $p_2 = 0.3$.

3. Two tensor case with 50 degree crossing and balanced components: $J = 2$, $\mathbf{m}_1 = (\cos(\pi/9), \sin(\pi/9), 0)^\intercal$, $\mathbf{m}_2 = (\sin(\pi/9), \cos(\pi/9), 0)^\intercal$, $p_1 = 0.5$, $p_2 = 0.5$.

All FAs and largest eigenvalues of underlying tensors are set to 0.9 and 4×10^{-3} respectively. Moreover, b, S_0 and σ are set to 1000, 1000 and 50 respectively. This has a signal-to-noise ratio ($\mathsf{SNR} := S_0/\sigma$) 20, which is typical for dMRI studies. $\mathcal{U}$ is obtained from the sphere tessellation with 3 subdivision using octahedron and $|\mathcal{U}| = 33$. For each setting, we simulate 200 voxel-wise data sets and compare the following methods:

- golden: Optimization of (2.4) via Broyden-Fletcher-Goldfarb-Shanno (BFGS) method with starting values set as the true parameter values. (J is known.)

- global-aic: Global optimization of (2.4) via GENOUD (Sekhon and Mebane, 1998) with Akaike Information criterion (AIC) for selection of J.

- global-bic: Similar to global-aic but with BIC.

- prop-aic: Our proposed method with AIC.

- prop-bic: Our proposed method with BIC.

Note that the AIC is derived as

$$\mathsf{AIC}(I) = -2l(\hat{\boldsymbol{\gamma}}(I)) + 8I.$$

The simulation results are summarized in Table A.1. With the information of true parameters, golden can be treated as a golden standard. Excluding golden, prop-bic has the highest proportion of correct estimation of J and attains around 99% correct recovery, which leads to our choice of BIC over AIC. In addition, note that prop-bic over-selects J when it does not estimate J correctly. This is one of the reasons why a

removal step (Step 12 of Algorithm 1) is designed in our smoothing procedure. As said, our goal is the diffusion direction $\mathbf{m}$. Conditional on the correct estimation of J, the squared error of $\mathbf{m}$ is defined as

$$\min_{\{k_1,\ldots,k_J\in\{1,\ldots,J\}:k_i\neq k_j\}} \sum_{j=1}^{J} d^{*2}(\mathbf{m}_j, \hat{\mathbf{u}}_{k_j}),$$

where $\hat{\mathbf{u}}_1, \ldots, \hat{\mathbf{u}}_J$ are the estimated diffusion directions. From Table A.1 all methods have root MSEs of $\mathbf{m}$ ranging from 1.5 to 1.6, 4.5 to 4.6 and 5.1 to 5.7 degree in the three settings respectively, and so these methods do not have big difference in terms of tracking. Given the accurate estimation of J and the computational benefit (over general global optimization methods), prop-bic performs the best among the compared methods.

Table A.1. Simulation results for voxel-wise estimation. **Correct-select**: proportion of $\hat{J} = J$. **Over-select**: proportion of $\hat{J} >= J$. $\mathbf{m}$, α, τ : MSE of $\mathbf{m}$, α and τ (computed on $\hat{J} = J$), with corresponding standard error stated in brackets. Note that the MSE of $\mathbf{m}$ is in squared degree.

Setting	Method	Correct-select	Over-select	$\mathbf{m}$	α	τ
1	golden	100%	100%	2.48 (3.06e-03)	5.70e-02 (4.66e-03)	2.69e-04 (2.32e-05)
	global-aic	75%	100%	2.39 (3.32e-03)	5.50e-02 (5.40e-03)	2.65e-04 (2.40e-05)
	global-bic	98%	100%	2.48 (3.11e-03)	5.65e-02 (4.70e-03)	2.67e-04 (2.33e-05)
	prop-aic	89%	100%	2.41 (3.07e-03)	5.53e-02 (5.01e-03)	2.73e-04 (2.51e-05)
	prop-bic	99.5%	100%	2.48 (3.07e-03)	5.65e-02 (4.65e-03)	2.69e-04 (2.33e-05)
2	golden	100%	100%	20.5 (1.99e-02)	1.23 (2.83e-01)	4.01e-04 (2.81e-05)
	global-aic	81.5%	100%	21.0 (2.19e-02)	1.20 (3.40e-01)	3.93e-04 (2.97e-05)
	global-bic	97%	100%	21.3 (2.07e-02)	1.92 (7.46e-01)	4.05e-04 (2.88e-05)
	prop-aic	91.5%	100%	21.1 (2.13e-02)	1.37 (3.43e-01)	4.10e-04 (2.93e-05)
	prop-bic	99.5%	100%	20.7 (2.01e-02)	1.33 (3.16e-01)	4.00e-04 (2.81e-05)
3	golden	100%	100%	28.7 (3.85e-02)	5.21 (3.24)	2.72e-03 (2.61e-04)
	global-aic	74.5%	100%	26.5 (3.80e-02)	2.02 (4.21e-01)	2.51e-03 (2.92e-04)
	global-bic	95.5%	100%	32.3 (7.20e-02)	3.38 (1.04)	2.95e-03 (3.47e-04)
	prop-aic	93.5%	100%	27.6 (3.83e-02)	5.37 (3.46)	2.60e-03 (2.65e-04)
	prop-bic	99%	100%	28.6 (3.86e-02)	5.23 (3.27)	2.70e-03 (2.62e-04)

A.3 Choice of bandwidth

This section presents our bandwidth selection methods for the smoothing method in Section 2.4. These methods are based on the idea of cross-validation (CV). Let $\breve{\mathbf{m}}_i^{-i}$ be the smoothed version of $\hat{\mathbf{m}}_i$ when all directions sharing the same voxel with $\hat{\mathbf{m}}_i$ are

not used in the smoothing. Since the choice of h may affect the number of clusters (steps 3 and 4 of Algorithm 1), $\hat{\mathbf{m}}_i$ may have been removed (step 12 of Algorithm 1). Thus, $\breve{\mathbf{m}}_i^{-i}$ is not always defined. Let o_i be the indicator of the existence of $\breve{\mathbf{m}}_i^{-i}$. The CV score is the mean of $\{d^{*2}(\hat{\mathbf{m}}_i, \breve{\mathbf{m}}_i^{-i}) : o_i = 1\}$.

Even after direction smoothing, the number of diffusion directions within a voxel may still be over-estimated. These spurious directions can have a great effect on the CV score, similar to the effect of outliers.

To alleviate this issue, the trimmed mean of $\{d^{*2}(\hat{\mathbf{m}}_i, \breve{\mathbf{m}}_i^{-i}) : o_i = 1\}$ and the median of $\{d^{*}(\hat{\mathbf{m}}_i, \breve{\mathbf{m}}_i^{-i}) : o_i = 1\}$ are used to form robust CV scores. They are called trimmed CV score and Median CV score respectively. We choose h as the minimizer of either one of these scores. See Section 2.7 for their numerical comparison.

In our numerical illustrations, the bandwidth h is chosen differently for single fiber regions and crossing fiber regions. Further, if one has enough computational resource, adaptive choice of bandwidth can also be achieved by dividing voxels into blocks according to their spatial locations and performing cross validation.

A.4 Algorithms

This section presents various algorithms developed in the main paper.

Algorithm 2 CLUSTDIR: PAM based clustering for direction vectors

Input: Set of direction vectors $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, number of cluster N_c

Output: Group mean $\{\mathbf{v}_1^*, \ldots, \mathbf{v}_{N_c}^*\}$, group label $\{e_1, \ldots, e_n\}$

procedure CLUSTDIR($\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, N_c)

  for $i, j = 1$ to n **do** $D_{ij} \leftarrow d^*(\mathbf{v}_i, \mathbf{v}_j)$

  Define $\mathbf{D}$ as the dissimilarity matrix with elements D_{ij}'s

  Apply PAM with dissimilarity matrix $\mathbf{D}$ to cluster $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$ into N_c groups

  for $i = 1$ to n **do** $e_i \leftarrow$ group label of $\mathbf{v}_i$

  for $j = 1$ to N_c **do** Compute group (Karcher) means:

$$v_j^* \leftarrow \arg\min_{\mathbf{v} \in \mathcal{M}} \sum_{i=1}^{n} I\{e_i = j\} d^{*2}(\mathbf{v}_i, \mathbf{v})$$

  return $(\{\mathbf{v}_1^*, \ldots, \mathbf{v}_{N_c}^*\}, \{e_1, \ldots, e_n\})$

Algorithm 3 Algorithm for voxel-wise estimation

Input: Observed signal intensities $\{S(\mathbf{u}), \mathbf{u} \in \mathcal{U}\}$, set of gradient vectors $\mathcal{U}$, non-diffusion weighted intensity S_0, standard deviation of the noise σ, b-value b, FA threshold r, upper bound of the number of directions $\tilde{I}$

Output: The selected number of diffusion directions, $\hat{J}$ and, if $\hat{J} > 0$, the corresponding ML estimate $\boldsymbol{\gamma}(\hat{J})$

Description: To perform voxel-wise estimation

Compute FA

if FA $< r$ **then**

  Declare there is no major diffusion direction: $\hat{J} \leftarrow 0$

else

  Estimate $\boldsymbol{\beta}$ (Appendix A.1) and determine the selected directions.

  for $I = 1, \ldots, \min\{\tilde{I}, L\}$ **do**

    Cluster the selected directions into I groups (Algorithm 2)

    Perform optimization with a gradient method (Section 2.3.3) and obtain ML estimate $\hat{\boldsymbol{\gamma}}(I)$

    Compute $\mathsf{BIC}(I)$

  Compute $\mathsf{BIC}(0)$

  Estimate the number of diffusion directions: $\hat{J} \leftarrow \text{argmin}_{I \in \{0, \ldots, \min\{\tilde{I}, L\}\}} \mathsf{BIC}(I)$

Algorithm 4 CLUSTDIRN: PAM based clustering algorithm for direction vectors with automatic choice of number of clusters

Input: Set of direction vector $\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, maximum number of cluster K, angular threshold ξ

Output: Group mean $\{\mathbf{v}_1^*, \ldots, \mathbf{v}_C^*\}$, number of clusters C

procedure CLUSTDIRN($\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, K, ξ)
  if $n = 1$ **then**
    the case of only one input direction: declare only one cluster
    $C \leftarrow 1$
  else if $n = 2$ **then**
    the case of two input directions: declare only one cluster if the angular separation of these directions are small
    if $d^*(\mathbf{v}_1, \mathbf{v}_2) \leq \xi$ **then** $C \leftarrow 1$ **else** $C \leftarrow 2$
  else if $n = 3$ **then**
    the case of three input directions
    $\psi \leftarrow$ the distance (2.7) between the two cluster means of CLUSTDIR($\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\}$, 2) (Algorithm 2)
    if $\psi \leq \xi$ **then**
      $C \leftarrow 1$
    else
      if minimum pairwise distance of $\{\mathbf{v}_1, \mathbf{v}_2, \mathbf{v}_3\} \leq \xi$ **then** $C \leftarrow 2$ **else** $C \leftarrow 3$
  else
    the case of more than three input directions: use Shilhouette criterion
    $\psi \leftarrow$ distance between the two cluster means of CLUSTDIR($\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, 2)
    if $\psi \leq \xi$ **then**
      Claim there is only one cluster if the angular separation is small: $C \leftarrow 1$
    else
      for $k = 2$ to K **do**
        $a_k \leftarrow$ average silhouette computed using CLUSTDIR($\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, k)
      Estimate the number of clusters as the maximizer of average silhouette: $C \leftarrow \arg\min_j \{a_j\}$
  $(\{\mathbf{v}_1^*, \ldots, \mathbf{v}_C^*\}, \{e_1, \ldots, e_n\}) \leftarrow$ CLUSTDIR($\{\mathbf{v}_1, \ldots, \mathbf{v}_n\}$, C)
  return $(\{\mathbf{v}_1^*, \ldots, \mathbf{v}_C^*\}, C)$

Algorithm 5 Algorithm for fiber tracking

Input: Target voxel $\mathbf{s}^*$, initial direction $\mathbf{v}^*$, (smoothed) voxel-wise estimate $\{(\mathbf{s}_k, \hat{\mathbf{v}}_k), k = 1, \ldots, T\}$, maximum number of projection N_{proj}, angular threshold ξ

Output: Recorded locations and directions

Description: To perform fiber tracking

1: Initialization: $\mathbf{x} \leftarrow \mathbf{s}^*$; $\mathbf{v} \leftarrow \mathbf{v}^*$; $Z \leftarrow$ True
Here, $\mathbf{x}$ represents the current location, $\mathbf{v}$ represents the current direction, Z is an indicator of whether the tracking should continue
2: Record $\mathbf{x}, \mathbf{v}$
3: **while** Z **do**
4: Move from $\mathbf{x}$ in the direction of $\mathbf{v}$ until hitting the boundary of the voxel
5: $\mathbf{x} \leftarrow$ boundary point of the voxel
6: $K \leftarrow$ number of fiber directions at the next voxel
7: **if** $K = 0$ **then**
8: $\tilde{Z} \leftarrow$ False, where $\tilde{Z}$ is an indicator of whether a viable direction exists
9: **else**
10: $\{\mathbf{v}_1, \ldots, \mathbf{v}_K\} \leftarrow$ fiber directions at the next voxel
11: Identify the direction with smallest angular separation: $j \leftarrow \arg\min_k d^*(\mathbf{v}, \mathbf{v}_k)$
12: **if** $d^*(\mathbf{v}, \mathbf{v}_j) \leq \xi$ **then**
13: $\mathbf{v} \leftarrow \text{sign}(\mathbf{v} \cdot \mathbf{v}_j)\mathbf{v}_j$; $\tilde{Z} \leftarrow$ True
14: **else**
15: $\tilde{Z} \leftarrow$ False
16: **if** not $\tilde{Z}$ **then**
Project the tracking and check if there is any viable direction after N_{proj} voxels:
17: $\tilde{\mathbf{x}} \leftarrow \mathbf{x}$; $\tilde{\mathbf{v}} \leftarrow \mathbf{v}$
18: **for** $n = 1$ to N_{proj} **do**
19: Projection: run lines 4 to 15 with all $\mathbf{x}$ and $\mathbf{v}$ replaced by $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{v}}$
20: **if** $\tilde{Z}$ **then**
21: Record $\tilde{\mathbf{x}}$ and $\tilde{\mathbf{v}}$; **break**
22: **if** not $\tilde{Z}$ **then** $Z \leftarrow$ False (Stop the tracking if there is no viable direction after N_{proj} voxels)
23: **else**
24: Record $\mathbf{x}, \mathbf{v}$

A.5 Technical details

Lemma A.1. *Assume that Assumption 1 hold.* $\psi(\boldsymbol{\omega}, \boldsymbol{\theta})$ *is twice continuously differentiable in a neighborhood of* $\boldsymbol{\theta}_0 = \mathbf{0}$, $m(s_0) = \mathbf{0}$ *and* $M_n^{(1)}(\mathbf{0}) = -2\sum_{i=1}^n hK_h(S_i - s_0)\boldsymbol{\theta}_i$.

Proof of Lemma A.1. Under Assumption 1, for $\boldsymbol{\theta}$ close to $\boldsymbol{\theta}_0$,

$$d(\boldsymbol{\theta}_i, \boldsymbol{\theta}) = \arccos(|\rho_{\mathbf{v}_0}(\mathbf{v}_i)^\intercal \phi^{-1}(\boldsymbol{\theta})|) = \arccos(\rho_{\mathbf{v}_0}(\mathbf{v}_i)^\intercal \phi^{-1}(\boldsymbol{\theta})).$$

Note that $\rho_{\mathbf{v}_0}(\mathbf{v}_i) \in \mathcal{V}$ is represented by $\boldsymbol{\theta}_i$. Thus, for $\boldsymbol{\theta}$ close to $\boldsymbol{\theta}_0$, $d(\boldsymbol{\theta}_i, \boldsymbol{\theta})$ coincides with the geodesic distance of $\mathcal{V}$ between points represented by logarithm coordinates $\boldsymbol{\theta}_i$ and $\boldsymbol{\theta}$. Now, Lemma A.1 follows from Bhattacharya and Bhattacharya (2012, Theorem 5.3) applied to the Manifold $\mathcal{V}$. Note that the cited theorem develops the coordinate system through the logarithm map at the intrinsic mean, which is not the same in our case. However, the requirement for developing the system at the intrinsic mean is for deeper results stated in their theorem, which is irrelevant to our use of their theorem. □

Lemma A.2. *Assume that Assumptions 1-5 hold. Let* $\mathbf{Y}_i = hK_h(S_i - s_0)m(S_i)$, *for* $i = 1, \ldots n$. *Then*

$$\sum_{i=1}^n \mathbf{Y}_i = nh^3 \int x^2 K(x)dx \left\{ m^{(1)}(s_0)f^{(1)}(s_0) + \frac{1}{2}m^{(2)}(s_0)f_S(s_0) \right\} + O_p(\sqrt{nh^3}),$$

where $m^{(1)}$ *and* $m^{(2)}$ *are interpreted as vectors of first and second derivatives of elements of* m *respectively.*

Proof of Lemma A.2. Since $\mathbf{Y}_i$'s are independently and identically distributed, we have

$$\sum_{i=1}^n \mathbf{Y}_i = n\mathbb{E}(\mathbf{Y}_1) + O_p\left\{ \sqrt{n\mathbb{E}(\mathbf{Y}_1^2)} \right\}.$$

We compute $\mathbb{E}(\mathbf{Y}_1)$ and $\mathbb{E}(\mathbf{Y}_1^2)$ below. Write $\mathbf{Y}_1 = (Y_{1,1}, Y_{1,2})^\intercal$. For $j = 1, 2$, by dominated convergence theorem with boundedness and continuity assumptions of f_S and m_j, and $m(s_0) = 0$, from Lemma A.1, we have

$$\begin{aligned}
\mathbb{E}(Y_{1,j}) &= \mathbb{E}\left\{hK_h(S_1 - s_0)m_j(S_1)\right\} \\
&= h\int K_h(s - s_0)m(s)f_S(s)ds \\
&= h\int K(x)m_j(s_0 + hx)f_S(s_0 + hx)dx \\
&= h\int K(x)\left\{m_j^{(1)}(s_0)hx + \frac{1}{2}m_j^{(2)}(s_0)h^2x^2\right\} \\
&\qquad \times \left\{f_S(s_0) + f_S^{(1)}(s_0)hx + \frac{1}{2}f_S^{(2)}(s_0)h^2x^2\right\}dx + o(h^3) \\
&= h^3\int x^2K(x)dx\left\{m_j^{(1)}(s_0)f_S^{(1)}(s_0) + \frac{1}{2}m_j^{(2)}(s_0)f_S(s_0)\right\} + o(h^3).
\end{aligned}$$

Similarly, for $j = 1, 2$,

$$\mathbb{E}(\mathbf{Y}_1^2) = \mathbb{E}\left\{h^2K_h^2(S_1 - s_0)m_j^2(S_i)\right\} = h^3\int x^2K^2(x)dx\{m_j^{(2)}(s_0)\}^2f_S(s_0) + o(h^3).$$

Thus,

$$\sum_{i=1}^n \mathbf{Y}_i = nh^3\int x^2K(x)dx\left\{m^{(1)}(s_0)f_S^{(1)}(s_0) + \frac{1}{2}m^{(2)}(s_0)f_S(s_0)\right\} + O_p(\sqrt{nh^3}).$$

□

Lemma A.3. *Assume that Assumptions 1-4 and 6 hold. Let* $\tilde{\mathbf{Y}}_i = hK_h(S_i - s_0)(\boldsymbol{\theta}_i - m(S_i))$, *for* $i = 1, \ldots n$. *Then*

$$\frac{1}{\sqrt{nh}}\sum_{i=1}^n \tilde{\mathbf{Y}}_i \implies \mathcal{N}_2\left(\mathbf{0}, \int K^2(x)dx f_S(s_0)\boldsymbol{\Sigma}(s_0)\right).$$

Proof of Lemma A.3. We will use the Linderberg-Feller central limit theorem for showing the asymptotic normality of $\sum_{i=1}^n \tilde{\mathbf{Y}}_i/\sqrt{nh}$. First, it is trivial that, for fixed n, $\tilde{\mathbf{Y}}_i$'s

are independently and identically distributed, with $\mathbb{E}(\tilde{\mathbf{Y}}_1) = 0$. Next, we study the variance of $\sum_{i=1}^{n} \tilde{\mathbf{Y}}_i/\sqrt{nh}$, which is $\mathbb{E}(\mathbf{Y}_1\mathbf{Y}_1^\intercal)/h$. Now, write $\mathbf{Y}_1 = (Y_{1,1}, Y_{1,2})^\intercal$. For $j,k = 1,2$,

$$\begin{aligned}
\frac{1}{h}\mathbb{E}(Y_{1,j}Y_{1,k}) &= h\int K_h^2(s-s_0)\mathbb{E}\left[\left\{\boldsymbol{\theta}_{1,j} - m_j(s_1)\right\}\left\{\boldsymbol{\theta}_{1,k} - m_k(s_1)\right\}|S_1 = s\right] f_S(s)ds \\
&= h\int K_h^2(s-s_0)\Sigma_{jk}(s)f_S(s)ds \\
&= \int K^2(x)f(s_0+hx)\Sigma_{jk}(s_0+hx)dx \\
&= \int K^2(x)dx f(s_0)\Sigma_{jk}(s_0) + o(1),
\end{aligned}$$

by dominated convergence theorem with boundedness and continuity assumptions of f_S and Σ_{jk}.

And, next, we have to verify the Linderberg-Feller condition. In our case, it can be reformulated as, for any $\varepsilon > 0$,

$$\lim_{n\to\infty}\sum_{i=1}^{n}\mathbb{E}\left(\left\|\frac{\tilde{\mathbf{Y}}_i}{\sqrt{nh}}\right\|^2 I\left\{\left\|\frac{\tilde{\mathbf{Y}}_i}{\sqrt{nh}}\right\| > \varepsilon\right\}\right) = 0.$$

We verify this condition by showing $\lim_{n\to\infty}\Pr(\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\| > \varepsilon) = 0$, for any $\varepsilon > 0$. This is equivalent to $\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\| = o_p(1)$, which we verify by looking at the second moment of $\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\|$.

$$
\begin{aligned}
\mathbb{E}\left(\left\|\frac{\tilde{\mathbf{Y}}_1}{\sqrt{nh}}\right\|^2\right) &= \frac{1}{nh}\mathbb{E}\left\{h^2K_h^2(S_1-s_0)\|\boldsymbol{\theta}_1-\mathbf{m}(S_1)\|^2\right\} \\
&= \frac{h}{n}\int K_h^2(s-s_0)\mathbb{E}(\|\boldsymbol{\theta}_1-\mathbf{m}(S_1)\|^2|S_1=s)f_S(s)ds \\
&= \frac{h}{n}\int K_h^2(s-s_0)\mathbb{E}(\text{trace}\left[\{\boldsymbol{\theta}_1-m(S_1)\}\{\boldsymbol{\theta}_1-\mathbf{m}(S_1)\}^\intercal\right]|S_1=s)f_S(s)ds \\
&= \frac{h}{n}\int K_h^2(s-s_0)\text{trace}\left\{\boldsymbol{\Sigma}(s)\right\}f_S(s)ds \\
&= \frac{1}{n}\int K(x)\text{trace}\left\{\boldsymbol{\Sigma}(s_0+hx)\right\}f_S(s_0+hx)dx \\
&= \frac{1}{n}\left[\{\Sigma_{11}(s_0)+\Sigma_{22}(s_0)\}f_S(s_0)+o(1)\right].
\end{aligned}
$$

Thus, $\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\| = o_p(1)$ and by continuous mapping theorem, $\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\|^2 = o_p(1)$.

$$
\sum_{i=1}^{n}\mathbb{E}\left(\left\|\frac{\tilde{\mathbf{Y}}_i}{\sqrt{nh}}\right\|^2 I\left\{\left\|\frac{\tilde{\mathbf{Y}}_i}{\sqrt{nh}}\right\|>\varepsilon\right\}\right) = \mathbb{E}\left(n\left\|\frac{\tilde{\mathbf{Y}}_1}{\sqrt{nh}}\right\|^2 I\left\{\left\|\frac{\tilde{\mathbf{Y}}_1}{\sqrt{nh}}\right\|>\varepsilon\right\}\right)
$$

Call the term inside the expectation of the right hand side as Z_n. From above, $\mathbb{E}(n\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\|^2)<\infty$, for sufficiently large n. Note that $Z_n \le n\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\|^2$. Thus, by dominated convergence theorem with application of Skorohod Representation Theorem to extend the result to weakly convergent sequence of random variables, we have $\lim_{n\to\infty}\mathbb{E}(n\|\tilde{\mathbf{Y}}_1/\sqrt{nh}\|^2)=0$ and thus Linderberg-Feller condition is verified. Hence, by Linderberg-Feller central limit theorem, we have

$$
\frac{1}{\sqrt{nh}}\sum_{i=1}^{n}\tilde{\mathbf{Y}}_i \implies \mathcal{N}_2\left(\mathbf{0}, \int K^2(x)dx f_S(s_0)\boldsymbol{\Sigma}(s_0)\right).
$$

□

Lemma A.4. *Assume that Assumption 1-4, 7 and 8 hold.*

$$
M_n^{(2)}(\boldsymbol{\theta}_0) = nh\boldsymbol{\Psi}(s_0)f_S(s_0)\{1+o_p(1)\}
$$

Proof of Lemma A.4. Note that $M_n^{(2)}(\boldsymbol{\theta}_0) = \sum_{i=1}^n hK_h(S_i - s_0)\boldsymbol{\psi}_2(\boldsymbol{\theta}_i, \boldsymbol{\theta}_0)$. To understand the asymptotic behavior of $M_n^{(2)}(\boldsymbol{\theta}_0)$, we study the asymptotic expansion of $hK_h(S_1 - s_0)\boldsymbol{\psi}_2(\boldsymbol{\theta}_i, \boldsymbol{\theta}_0)$ through computing its first two moments.

For $j, k = 1, 2$,

$$\begin{aligned}\mathbb{E}\left\{hK_h(S_1 - s_0)[\boldsymbol{\psi}_2(\boldsymbol{\theta}_1, \boldsymbol{\theta}_0))]_{j,k}\right\} &= \int hK_h(s - s_0)\Psi_{jk}(s)f_S(s)ds \\ &= h\left\{\Psi_{jk}(s_0)f_S(s_0) + o(1)\right\},\end{aligned}$$

by dominated convergence theorem with boundedness and continuity assumptions of f_S and Ψ_{jk}. As the second moment, since $E\{[\boldsymbol{\psi}_2(\boldsymbol{\theta}_1, \boldsymbol{\theta}_0)]_{j,k}^2 | S_1 = s\}$ is bounded,

$$\begin{aligned}\mathbb{E}\left\{h^2K_h^2(S_1 - s_0)[\boldsymbol{\psi}_2(\boldsymbol{\theta}_1, \boldsymbol{\theta}_0))]_{j,k}^2\right\} &\leq C_{jk}\int h^2K_h^2(s - s_0)f_S(s)ds \\ &= h\left\{C_{jk}f_S(s_0)\int K^2(x)dx + o(1)\right\}\end{aligned}$$

by dominated convergence theorem with boundedness and continuity of f_S. Thus,

$$M_n^{(2)}(\boldsymbol{\theta}_0) = nh\boldsymbol{\Psi}(s_0)f_S(s_0)\{1 + o_p(1)\}.$$

□

Lemma A.5. *Assume that Assumptions 1-4 and 7-10 hold. Let $\boldsymbol{\theta} \in \mathbb{R}^2$. For all sufficiently small $\delta > 0$,*

$$\begin{aligned}\lim_{n\to\infty}\Pr\left[\inf_{\tilde{\boldsymbol{\theta}}\in\mathcal{B}_\delta(\boldsymbol{\theta}_0)}\frac{1}{nh}\left\{(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal M_n^{(2)}(\tilde{\boldsymbol{\theta}})(\boldsymbol{\theta} - \boldsymbol{\theta}_0)\right\} \geq \frac{1}{2}f_S(s_0)(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal\boldsymbol{\Psi}(s_0)(\boldsymbol{\theta} - \boldsymbol{\theta}_0)\right] \\ = 1.\end{aligned}$$

Proof of Lemma A.5. In this proof, we will prepare the uniform result that is required to show consistency of our estimator. Write $\mathbf{T}_n(\tilde{\boldsymbol{\theta}}) = (1/n)\sum_{i=1}^n K_h(S_i - s_0)\left\{\boldsymbol{\psi}_2(\boldsymbol{\theta}_i, \tilde{\boldsymbol{\theta}}) -\right.$

$\psi_2(\boldsymbol{\theta}_i, \boldsymbol{\theta}_0)$. Note that

$$\sup_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \|\mathbf{T}_n(\tilde{\boldsymbol{\theta}})\| \leq \frac{1}{n} \sum_{i=1}^{n} \left\{ K_h(S_i - s_0) \sup_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \left\| \psi_2(\boldsymbol{\theta}_i, \tilde{\boldsymbol{\theta}}) - \psi_2(\boldsymbol{\theta}_i, \boldsymbol{\theta}_0) \right\| \right\}.$$

By dominated convergence theorem and Assumption 9, we have

$$\begin{aligned}
\mathbb{E}\left(\sup_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \|\mathbf{T}_n(\tilde{\boldsymbol{\theta}})\| \right) &\leq \int K_h(s - s_0) \gamma(\delta, s) f_S(s) ds \\
&= \int K(x) \gamma(\delta, s_0 + hx) f_S(s_0 + hx) dx \\
&\leq \tilde{\gamma}(\delta) f(s_0) + o(1).
\end{aligned}$$

By Assumption 9, we have $\lim_{\delta \to 0} \operatorname{limsup}_{n \to \infty} [\sup_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}(\boldsymbol{\theta}_0)} \|\mathbf{T}_n(\tilde{\boldsymbol{\theta}})\|] = 0$ in probability. For a given $\boldsymbol{\theta} \in \mathbb{R}^2$, note that

$$\sup_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \left| (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal \mathbf{T}_n(\tilde{\boldsymbol{\theta}}) (\boldsymbol{\theta} - \boldsymbol{\theta}_0) \right| \leq \left(\sup_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \|\mathbf{T}_n(\tilde{\boldsymbol{\theta}})\| \right)^2 \|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|^2.$$

Thus, by Lemma A.4 and Assumption 10, for all sufficiently small $\delta > 0$,

$$\begin{aligned}
\lim_{n \to \infty} \Pr \left[\inf_{\tilde{\boldsymbol{\theta}} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \frac{1}{nh} \left\{ (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal M_n^{(2)}(\tilde{\boldsymbol{\theta}}) (\boldsymbol{\theta} - \boldsymbol{\theta}_0) \right\} \geq \frac{1}{2} f_S(s_0) (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal \boldsymbol{\Psi}(s_0) (\boldsymbol{\theta} - \boldsymbol{\theta}_0) \right] \\
= 1.
\end{aligned}$$

□

Proof of Theorem 2.1(a). To show the consistency result, we look into the Taylor's expansion of $M_n(\boldsymbol{\theta})$ around $\boldsymbol{\theta}_0$. Consider $\boldsymbol{\theta} \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)$ and by Taylor's expansion, we have

$$M_n(\boldsymbol{\theta}) - M_n(\boldsymbol{\theta}_0) = M_n^{(1)}(\boldsymbol{\theta}_0)^\intercal (\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \frac{1}{2} (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal M_n^{(2)}(\boldsymbol{\theta}^*) (\boldsymbol{\theta} - \boldsymbol{\theta}_0),$$

where $\boldsymbol{\theta}^*$ lies on the line segment joining $\boldsymbol{\theta}_0$ and $\boldsymbol{\theta}$. First, by Lemma A.2 and A.3,

$$\frac{1}{nh} M_n^{(1)}(\boldsymbol{\theta}_0) = -\frac{2}{nh} \sum_{i=1}^{n} \left(\mathbf{Y}_i + \tilde{\mathbf{Y}}_i \right) = o_p(1).$$

Then, from Lemma A.5, we have, for all sufficiently small $\delta > 0$,

$$\lim_{n\to\infty} \Pr \left[\inf_{\boldsymbol{\theta}^* \in \mathcal{B}_\delta(\boldsymbol{\theta}_0)} \frac{1}{nh} \left\{ (\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal M_n^{(2)}(\boldsymbol{\theta}^*)(\boldsymbol{\theta} - \boldsymbol{\theta}_0) \right\} \geq \frac{1}{2} f_S(s_0)(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal \boldsymbol{\Psi}(s_0)(\boldsymbol{\theta} - \boldsymbol{\theta}_0) \right] = 1.$$

Thus, by Assumption 10, there exists a local minimum in $\mathcal{B}_\delta(\boldsymbol{\theta}_0)$ asymptotically. That means, for any $\delta > 0$, there exists a sequence of roots, $\hat{\boldsymbol{\theta}}_n$, to $M_n^{(1)}(\boldsymbol{\theta}) = 0$ such that,

$$\lim_{n\to\infty} \Pr(\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| < \delta) = 1.$$

This completes the proof of Theorem 2.1(a). □

Proof of Theorem 2.1(b). To show the distributional result, we expand $M_n^{(1)}(\boldsymbol{\theta})$ by Taylor's expansion. Expanding at $\hat{\boldsymbol{\theta}}_n$, stated in Theorem 2.1(b),

$$0 = M_n^{(1)}(\hat{\boldsymbol{\theta}}_n) = M_n^{(1)}(\boldsymbol{\theta}_0) + M_n^{(2)}(\boldsymbol{\theta}_n^*)(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0),$$

where $\boldsymbol{\theta}_n^*$ lies on the line segment joining $\boldsymbol{\theta}_0$ and $\hat{\boldsymbol{\theta}}_n$. Note that $\|\boldsymbol{\theta}_n^* - \boldsymbol{\theta}_0\| \leq \|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| = o_p(1)$. And

$$\frac{1}{nh} M_n^{(2)}(\boldsymbol{\theta}_n^*) = f_S(s_0)\boldsymbol{\Psi}(s_0)\{1 + o_p(1)\}$$

since, with $\|\boldsymbol{\theta}_n^* - \boldsymbol{\theta}_0\| = o_p(1)$, one can show that $\mathbb{E}\{\mathbf{T}_n(\boldsymbol{\theta}_n^*)\} = o_p(1)$ along the proof of Lemma A.5. As for $M_n^{(1)}(\boldsymbol{\theta}_0)$, by Lemma A.2,

$$\begin{aligned} \frac{1}{\sqrt{nh}} M_n^{(1)}(\boldsymbol{\theta}_0) = (-2)\sqrt{nh^5} \int x^2 K(x) dx \left\{ m^{(1)}(s_0) f^{(1)}(s_0) + \frac{1}{2} m^{(2)}(s_0) f_S(s_0) \right\} \\ + (-2) \frac{1}{\sqrt{nh}} \sum_{i=1}^{n} \tilde{\mathbf{Y}}_i + o_p(1) \end{aligned}$$

Thus, by Slutsky's theorem,

$$\sqrt{nh}\left\{(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) - h^2\boldsymbol{\eta}\right\} \implies \mathcal{N}_2(\mathbf{0}, \boldsymbol{\Omega}).$$

□

Appendix B

Supplement to "A Frequentist Approach to Computer Model Calibrations"

B.1 Technical details

Lemma B.1 (Consistency of $\hat{\boldsymbol{\theta}}_n$)**.** *Assume that Assumptions 4.1, 4.2, 4.3(a), 4.4(a), 4.5(a), 4.6 hold.* $\hat{\boldsymbol{\theta}}_n$ *is a consistent estimator of* $\boldsymbol{\theta}_0$*. i.e.* $\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| \xrightarrow{P} 0$ *as* $n \to \infty$*.*

Proof of Lemma B.1. Note that

$$M_n(\boldsymbol{\theta}) = \frac{1}{n}\sum_{i=1}^{n} \varepsilon_i^2 + \|\zeta - g_{\boldsymbol{\theta}}\|_n^2 + 2\langle \varepsilon, \zeta - g_{\boldsymbol{\theta}} \rangle_n.$$

Define

$$M_{0,n}(\boldsymbol{\theta}) = \|\zeta - g_{\boldsymbol{\theta}}\|_n^2 + \sigma^2.$$

In order to derive a uniform convergence of $M_n(\boldsymbol{\theta}) - M_{0,n}(\boldsymbol{\theta})$, we need a uniform results on $\langle \varepsilon, \zeta - g_{\boldsymbol{\theta}} \rangle_n$. Here, we borrow the result from Corollary 8.3 of Van De Geer (2000). By Assumptions 4.2 and 4.3(a),

$$H(u, \mathcal{G} - \zeta, F_n) \leq d \log\left(\frac{4R_1 c_0 + u}{u}\right),$$

where H is the entropy (see Van De Geer, 2000). Thus the entropy integral converges:

$$\int_0^1 H^{1/2}(u, \mathcal{G} - \zeta, F_n) du < \infty.$$

Thus, using Corollary 8.3 of Van De Geer (2000) with Assumptions 4.1 and 4.4(a), we have

$$\sup_{\boldsymbol{\theta}\in\Theta} |\langle \varepsilon, \zeta - g_{\boldsymbol{\theta}} \rangle_n| = \mathcal{O}_p(1).$$

By Bernstein's inequality, we have $(1/n)\sum_{i=1}^n \varepsilon_i^2 \xrightarrow{P} \sigma^2$ and thus $\sup_{\boldsymbol{\theta}\in\Theta} |M_n(\boldsymbol{\theta}) - M_{0,n}(\boldsymbol{\theta})| = o_p(1)$. Consider

$$\sup_{\boldsymbol{\theta}\in\Theta} |M_n(\boldsymbol{\theta}) - M(\boldsymbol{\theta})| \leq \sup_{\boldsymbol{\theta}\in\Theta} |M_n(\boldsymbol{\theta}) - M_{0,n}(\boldsymbol{\theta})| + \sup_{\boldsymbol{\theta}\in\Theta} |M_{0,n}(\boldsymbol{\theta}) - M(\boldsymbol{\theta})|,$$

where $M(\boldsymbol{\theta}) = \|\zeta - g_{\boldsymbol{\theta}}\|^2 + \sigma^2$. By Assumption 4.5(a), $\sup_{\boldsymbol{\theta}\in\Theta} |M_n(\boldsymbol{\theta}) - M(\boldsymbol{\theta})| = o_p(1)$. By Theorem 5.7 of Van der Vaart (2000) with Assumption 4.6, $\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| = o_p(1)$. □

Proof of Theorem 4.1. We first derive a basic inequality. As $g_{\hat{\boldsymbol{\theta}}_n}$ minimizes $M_n(\boldsymbol{\theta})$,

$$\|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n^2 \leq 2\langle \varepsilon, g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0} \rangle_n + 2\langle \zeta - g_{\boldsymbol{\theta}_0}, g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0} \rangle_n. \tag{B.1}$$

The first term in the left hand side can be handled by the following result:

$$\frac{\langle \zeta - g_{\boldsymbol{\theta}_0}, g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0} \rangle_n}{\|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n} = \mathcal{O}_p(n^{-1/2}) \tag{B.2}$$

The proof of this result is shown in part of the proof of Theorem 9.1 of Van De Geer (2000). Define $\mathcal{G}_n(R) = \{g_{\boldsymbol{\theta}} \in \mathcal{G} : \|g_{\boldsymbol{\theta}} - g_{\boldsymbol{\theta}_0}\| \leq R\}$. By Assumptions 4.2 and 4.3(a), for $0 \leq z \leq 1$,

$$\begin{aligned}
\int_0^z H^{1/2}(u, \mathcal{G}_n(z), F_n) du &\leq \int_0^z d^{1/2} \left\{ \log\left(\frac{4c_0 z + u}{u} \right) \right\}^{1/2} du \\
&= 4c_0 d^{1/2} z \int_0^{1/4c_0} \left\{ \log\left(\frac{1}{\boldsymbol{x}} + 1 \right) \right\}^{1/2} d\boldsymbol{x} \\
&= K d^{1/2} z,
\end{aligned}$$

for some constant K. Take $\Psi(z) = K d^{1/2} z$. Thus, condition (9.3) of Van De Geer (2000) is met. In addition, the entropy integral converges:

$$\int_0^z H^{1/2}(u, \mathcal{G} - g_{\boldsymbol{\theta}_0}, F_n) du < \infty.$$

Then (B.2) follows from the proof of Theorem 9.1 of Van De Geer (2000) via the peeling device. (Note that the statement of Theorem 9.1 of Van De Geer (2000) requires $\Psi(z)/z^2$ to be non-decreasing, which is a typo. Rather, it should be non-increasing.)

The major difficulty lies in the second term of left hand side of (B.1), which has the same convergence rate as the left hand side (see below). This second term arises from the misspecification of the regression function, which results in non-mean-zero errors $(\delta_0(\boldsymbol{x}_i) + \varepsilon_i)$. This forbids us from the direct use of the inequality (B.1) for getting the convergence rate of $\|\boldsymbol{\theta} - \boldsymbol{\theta}_0\|$, as a standard approach shown in Van De Geer (2000).

From Assumptions 4.3(b) and 4.6, as $\boldsymbol{\theta}_0$ minimizes $M(\boldsymbol{\theta})$,

$$\int_{\mathcal{X}} \delta_0(\boldsymbol{x}) g^{(1)}_{\boldsymbol{\theta}_0}(\boldsymbol{x}) dF(\boldsymbol{x}) = 0$$

and $A = A_1 - A_2$ is strictly positive definite, where

$$A_1 = \int_{\mathcal{X}} g^{(1)}_{\boldsymbol{\theta}_0}(\boldsymbol{x}) g^{(1)}_{\boldsymbol{\theta}_0}(\boldsymbol{x})^\intercal dF(\boldsymbol{x}) \qquad \text{and} \qquad A_2 = \int_{\mathcal{X}} \delta_0(\boldsymbol{x}) g^{(2)}_{\boldsymbol{\theta}_0}(\boldsymbol{x}) dF(\boldsymbol{x}).$$

By Taylor's expansion, we also have, for $\boldsymbol{\theta} \in \Theta$ close to $\boldsymbol{\theta}_0$ and $\boldsymbol{x} \in \mathcal{X}$,

$$g_{\boldsymbol{\theta}}(\boldsymbol{x}) = g_{\boldsymbol{\theta}_0}(\boldsymbol{x}) + g^{(1)}_{\boldsymbol{\theta}_0}(\boldsymbol{x})^\intercal(\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal g^{(2)}_{\boldsymbol{\theta}_0}(\boldsymbol{x})(\boldsymbol{\theta} - \boldsymbol{\theta}_0) + \gamma_{\boldsymbol{\theta}}(\boldsymbol{x}), \qquad \text{(B.3)}$$

where

$$\gamma_{\boldsymbol{\theta}}(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{\theta} - \boldsymbol{\theta}_0)^\intercal \{g^{(2)}_{\tilde{\boldsymbol{\theta}}}(\boldsymbol{x}) - g^{(2)}_{\boldsymbol{\theta}_0}(\boldsymbol{x})\}(\boldsymbol{\theta} - \boldsymbol{\theta}_0).$$

Here $\tilde{\boldsymbol{\theta}}$ lies between $\boldsymbol{\theta}$ and $\boldsymbol{\theta}_0$. Now, by Assumption 4.3(c) and 4.7(a), and Lemma B.1, we have

$$\langle \delta_0, g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0} \rangle = \frac{1}{2}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0)^\intercal A_2 (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) + o_p(\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\|^2).$$

By Assumption 4.3(c) and Lemma B.1,

$$\|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n^2 = (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0)^\intercal A_{1,n} (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) + o_p(\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\|^2),$$

where $A_{1,n} = (1/n)\sum_{i=1}^n g^{(1)}_{\boldsymbol{\theta}_0}(\boldsymbol{x}_i) g^{(1)}_{\boldsymbol{\theta}_0}(\boldsymbol{x}_i)^\intercal$. Given Assumption 4.5(b) and 4.5(c), and (B.2), (B.1) becomes

$$\begin{aligned}(\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0)^\intercal A_{1,n} (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) &\le (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0)^\intercal A_2 (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) + o_p(\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\|^2) \\ &\quad + \mathcal{O}_p(n^{-1/2}) \|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n \\ (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0)^\intercal A (\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0) &\le o_p(\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\|^2) + \mathcal{O}_p(n^{-1/2}) \|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n\end{aligned}$$

Write the smallest eigenvalue of A be a. Since A is strictly positive definite, $a > 0$. Thus,

$$0 \leq a\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\|^2 \leq o_p(\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\|^2) + \mathcal{O}_p(n^{-1/2})\|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n.$$

And by Assumption 4.3(a),

$$a + o_p(1) \leq \frac{\mathcal{O}_p(n^{-1/2})}{\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\|},$$

which implies $\|\hat{\boldsymbol{\theta}}_n - \boldsymbol{\theta}_0\| = \mathcal{O}_p(n^{-1/2})$. By Assumption 4.3(a),

$$\|g_{\hat{\boldsymbol{\theta}}_n} - g_{\boldsymbol{\theta}_0}\|_n = \mathcal{O}_p(n^{-1/2}).$$

□

Lemma B.2. *Assume that ε_i's are uniformly sub-gaussian random variables and z_n is a function of $\boldsymbol{x} \in \mathcal{X}$ such that $\|z_n\|_n = \mathcal{O}_p(n^{-1/2})$. Moreover, assume that Assumption 4.7(b) holds. Let*

$$\tilde{\delta}_n = \arg\min_{\delta \in \mathcal{H}} \left[\frac{1}{n} \sum_{i=1}^{n} \{\tilde{y}_i - \delta(\boldsymbol{x}_i)\}^2 + \lambda_n^2 J^v(\delta) \right], \tag{B.4}$$

$\tilde{y}_i = \delta_0(\boldsymbol{x}_i) + z_n(\boldsymbol{x}_i) + \varepsilon_i$ for $i = 1, \ldots, n$. Suppose $v > (2\alpha)/(2+\alpha)$ and $\lambda_n \asymp n^{-1/(2+\alpha)}$.

(i) If $J(\delta_0) > 0$, we have

$$\|\tilde{\delta}_n - \delta_0\|_n = \mathcal{O}_p\left(n^{-1/(2+\alpha)}\right).$$

(ii) If $J(\delta_0) = 0$, $J(\delta) > 0$ for all $\delta \in \mathcal{H}$, and $4v < (2+\alpha)(2v - 2\alpha + v\alpha)$, we have

$$\|\tilde{\delta}_n - \delta_0\|_n = \mathcal{O}_p\left(n^{-1/2}\right).$$

Proof of Lemma B.2. The proof is similar to the proof of Theorem 10.2 of Van De Geer (2000), with modification to cope with the contamination z_n.

Case (i): Suppose $J(\tilde{\delta}_n) > J(\delta_0)$. Since $\tilde{\delta}_n$ minimizes (B.4), combining with Assumption 4.7(b) and Cauchy-Schwarz inequality,

$$\begin{aligned} \|\tilde{\delta}_n - \delta_0\|_n^2 + \lambda_n^2 J^v(\tilde{\delta}_n) \leq \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n^{1-\alpha/2} J^{\alpha/2}(\tilde{\delta}_n) + \lambda_n^2 J^v(\delta_0) \\ + \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n. \end{aligned} \tag{B.5}$$

Now, we look at two cases (a) $J(\delta_0) = 0$ and (b) $J(\delta_0) > 0$.

Case (i)(a): Suppose $J(\delta_0) = 0$. (B.5) becomes

$$\|\tilde{\delta}_n - \delta_0\|_n^2 + \lambda_n^2 J^v(\tilde{\delta}_n) \leq \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n^{1-\alpha/2} J^{\alpha/2}(\tilde{\delta}_n) + \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n.$$

Either

$$\|\tilde{\delta}_n - \delta_0\|_n^2 + \lambda_n^2 J^v(\tilde{\delta}_n) \leq \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n, \tag{B.6}$$

or

$$\|\tilde{\delta}_n - \delta_0\|_n^2 + \lambda_n^2 J^v(\tilde{\delta}_n) \leq \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n^{1-\alpha/2} J^{\alpha/2}(\tilde{\delta}_n). \tag{B.7}$$

Both (B.6) and (B.7) lead to $\|\tilde{\delta}_n - \delta_0\|_n = \mathcal{O}_p(n^{-1/2})$.

Case (i)(b): Suppose $J(\delta_0) > 0$. Let $\mathcal{A}_n$ be the event that the last term of (B.5) is of the largest term of the right hand side of (B.5). On $\mathcal{A}_n$, we have $\lambda_n^2 \leq \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n$ and

$$\|\tilde{\delta}_n - \delta_0\|_n^2 + \lambda_n^2 J^v(\tilde{\delta}_n) \leq \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n,$$

which leads to $\|\tilde{\delta}_n - \delta_0\|_n \leq \mathcal{O}_p(n^{-1/2})$. Thus, $\lambda_n \leq \mathcal{O}_p(n^{-1/2})$. However, $\lambda_n \asymp n^{-1/(2+\alpha)}$. Thus, $\Pr(\mathcal{A}_n) \to 0$ as $n \to \infty$. Lemma B.2 follows from the proof of Theorem 10.2 of Van De Geer (2000) by focusing on $\mathcal{A}_n^c$.

Case (ii): Suppose $J(\tilde{\delta}_n) \leq J(\delta_0)$ and $J(\delta_0) > 0$. For this case, we have

$$\|\tilde{\delta}_n - \delta_0\|_n^2 \leq \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n^{1-\alpha/2} J^{\alpha/2}(\delta_0) + \lambda_n^2 J^v(\delta_0) + \mathcal{O}_p(n^{-1/2})\|\tilde{\delta}_n - \delta_0\|_n. \tag{B.8}$$

Let $\mathcal{B}_n$ be the event that the last term of (B.8) is of the largest term of the right hand side of (B.8). Using similar argument of $\mathcal{A}_n$, we can show that $\Pr(\mathcal{B}_n) \to 0$ as $n \to \infty$. The rest follows from the proof of Theorem 10.2 of Van De Geer (2000) by looking into $\mathcal{B}_n^c$. □

Proof of Theorem 4.2. This follows from Theorem 4.1 and Lemma B.2. □

Proof of Corollary 4.1. The key idea is the same as Section 10.1.1 of Van De Geer (2000) by rewriting $\delta = \delta_1 + \delta_2$ for $\delta \in \mathcal{H}$, where $\delta_1 = \sum_{k=1}^m \psi_k$ and $\delta_2 = \int_0^1 \beta_u \tilde{\phi}_u$ such that $\langle \psi_k, \tilde{\phi}_u \rangle_n = 0$ for $k = 1, \ldots, m$ and $0 < u \leq 1$. One choice of $\{\psi_k\}$ and $\{\tilde{\phi}_u\}$ can be found in Example 9.3.2 of Van De Geer (2000).

Now, $\hat{\delta}_n$ can be estimated via two separate estimations. To see that, we write the least square criterion in (4.3) as

$$\|y - g_{\hat{\boldsymbol{\theta}}_n} - \delta\|_n^2 = \|y - g_{\hat{\boldsymbol{\theta}}_n} - \delta_0\|_n + \|\delta_0 - \delta\|_n^2 + 2\langle y - g_{\hat{\boldsymbol{\theta}}_n} - \delta_0, \delta_0 - \delta\rangle_n.$$

Here, the first term is a constant with respect to δ. The second term can be written as

$$\|\delta_0 - \delta\|_n^2 = \|\delta_{0,1} - \delta_1\|_n^2 + \|\delta_{0,2} - \delta_2\|_n^2$$

where

$$\delta_{0,1} \in \mathcal{H}_1 = \{\sum_{k=1}^{m} \alpha_k \psi_k : \alpha_k \in \mathbb{R}\} \tag{B.9}$$

and

$$\delta_{0,2} \in \mathcal{H}_2 = \mathcal{H} \setminus \mathcal{H}_1.$$

And

$$\langle y - g_{\hat{\boldsymbol{\theta}}_n} - \delta_0, \delta_0 - \delta\rangle_n = \langle \varepsilon + g_{\boldsymbol{\theta}_0} - g_{\hat{\boldsymbol{\theta}}_n}, \delta_{0,1} - \delta_1\rangle_n + \langle \varepsilon + g_{\boldsymbol{\theta}_0} - g_{\hat{\boldsymbol{\theta}}_n}, \delta_{0,2} - \delta_2\rangle_n.$$

The estimator can be written as $\hat{\delta}_n = \hat{\delta}_{1,n} + \hat{\delta}_{2,n}$, where

$$\begin{aligned}
\hat{\delta}_{1,n} &= \arg\min_{\delta_1 \in \mathcal{H}_1} \left\{ \|\delta_1 - \delta_{0,1}\|_n^2 - 2\langle \varepsilon + g_{\boldsymbol{\theta}_0} - g_{\hat{\boldsymbol{\theta}}_n}, \delta_1 - \delta_{0,1}\rangle_n \right\} \\
\hat{\delta}_{2,n} &= \arg\min_{\delta_2 \in \mathcal{H}_2} \left\{ \|\delta_2 - \delta_{0,2}\|_n^2 - 2\langle \varepsilon + g_{\boldsymbol{\theta}_0} - g_{\hat{\boldsymbol{\theta}}_n}, \delta_2 - \delta_{0,2}\rangle_n + \lambda_n^2 J^2(\delta_2) \right\}.
\end{aligned}$$

As for $\hat{\delta}_{1,n}$, by Theorem 4.1, we have

$$\begin{aligned}
\|\hat{\delta}_{1,n} - \delta_{0,1}\|_n^2 &\leq 2\langle \varepsilon, \hat{\delta}_{1,n} - \delta_{0,1}\rangle_n + 2\langle g_{\boldsymbol{\theta}_0} - g_{\hat{\boldsymbol{\theta}}_n}, \hat{\delta}_{1,n} - \delta_{0,1}\rangle_n \\
&\leq 2\langle \varepsilon, \hat{\delta}_{1,n} - \delta_{0,1}\rangle_n + \mathcal{O}_p(n^{-1/2})\|\hat{\delta}_{1,n} - \delta_{0,1}\|_n
\end{aligned}$$

Applying Theorem 9.1 of Van De Geer (2000), $\|\hat{\delta}_{1,n} - \delta_{0,1}\|_n = \mathcal{O}_p(n^{-1/2})$. As for $\hat{\delta}_{2,n}$, we simply apply Lemma B.2. Note that with smallest eigenvalue of $\int \psi\psi^\intercal dF_n$ bounded away from zero, Assumption 4.7(b) is fulfilled for $\mathcal{H}_2$ (Mammen, 1991). Then the corollary follows. □

Appendix C

Supplement to "Automatic Estimation of Flux Distributions of Astrophysical Source Populations"

C.1 Technical Details

To prove Theorem 5.1, the five assumptions made in Section 2 of Kiefer and Wolfowitz (1956) need to be verified. This is done in the following.

Assumption C.1. *It is required that $f(y;\boldsymbol{\theta})$ is a density with respect to a σ-finite measure μ on a Euclidean space of which y is the generic point.*

Proof. This condition is satisfied since the underlying distribution is discrete. □

Define a metric on the space Θ by setting

$$\delta(\boldsymbol{\theta}_1, \boldsymbol{\theta}_2) = \sum_{j=1}^{B} |\arctan \beta_{1,j} - \arctan \beta_{2,j}| + \sum_{j=1}^{B} |\arctan \tau_{1,j} - \arctan \tau_{2,j}|.$$

Following Kiefer and Wolfowitz (1956), the parameter space is compactified by defining $\bar{\Theta}$ to be the completion of Θ by adding all the limits of its Cauchy sequences in the sense of the above metric. Unless otherwise mentioned, all limits involving $\boldsymbol{\theta}$ are understood to be with respect to δ. The Euclidean norm is denoted by $|\cdot|_E$. To verify the next assumption of Kiefer and Wolfowitz (1956), two auxiliary lemmas are introduced.

Lemma C.1. *For sufficiently large β_j and a fixed $y \in \mathbb{N}_0$, we have*

$$\beta_j (A\tau_j)^{\beta_j} \int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} dt < 2(A\tau_j)^y e^{-A\tau_j},$$

where $\boldsymbol{\tau} \in \Theta$.

Proof. Note that $\beta_j (A\tau_j)^{\beta_j} \int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} dt \leq \beta_j (A\tau_j)^{\beta_j} \Gamma(y-\beta_j, A\tau_j)$. Thus, by Theorem 2.2 of Borwein and Chan (2009), for a sufficiently large β_j and a fixed $y \in \mathbb{N}_0$,

$$\Gamma(y-\beta_j, A\tau_j) \leq \frac{-(A\tau_j)^{y-\beta_j} e^{-A\tau_j}}{y-\beta_j}$$

and consequently $\beta_j (A\tau_j)^{\beta_j} \Gamma(y-\beta_j, A\tau_j) < 2(A\tau_j)^y e^{-A\tau_j}$. □

Lemma C.2. *If $|\boldsymbol{\tau}|_E < \infty$, $\lim_{|\boldsymbol{\beta}|_E \to \infty} f(y;\boldsymbol{\theta})$ exists.*

Proof. Note that if $|\boldsymbol{\beta}|_E \to \infty$, there exists a j such that $\beta_j \to \infty$. We focus on that one particular j. Let $g_j(y;\boldsymbol{\theta}) = \beta_j (A\tau_j)^{\beta_j} \int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} dt$. In order to show that the limit of f exists, we only have to show that the limit of g exists (since it generalizes to any j with $\beta_j \to \infty$). Note that, instead of considering g_j, we look at $h_j(y;\boldsymbol{\theta}) = \log g_j(y;\boldsymbol{\theta})$. We define $h_{j,1}(y;\boldsymbol{\theta}) = \log\{\beta_j (A\tau_j)^{\beta_j}\}$ and $h_{j,2}(y;\boldsymbol{\theta}) = \log \int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} dt$. Then we have

$$\begin{aligned}
\frac{\partial h_j(y;\boldsymbol{\theta})}{\partial \beta_j} &= \frac{\partial h_{j,1}(y;\boldsymbol{\theta})}{\partial \beta_j} + \frac{\partial h_{j,2}(y;\boldsymbol{\theta})}{\partial \beta_j} \\
&= \left\{ \frac{1}{\beta_j} + \log(A\tau_j) \right\} + \left\{ -\frac{\int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} (\log t) dt}{\int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} dt} \right\} \\
\frac{\partial^2 h_{j,1}(y;\boldsymbol{\theta})}{\partial \beta_j^2} &= -\frac{1}{\beta_j^2} \\
\frac{\partial^2 h_{j,2}(y;\boldsymbol{\theta})}{\partial \beta_j^2} &= \frac{\int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} (\log t)^2 dt}{\int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} dt} - \left\{ \frac{\int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} (\log t) dt}{\int_{A\tau_j}^{A\tau_{j+1}} t^{y-\beta_j-1} e^{-t} dt} \right\}^2
\end{aligned}$$

Note that $\partial^2 h_{j,2}(y;\boldsymbol{\theta}) / \partial \beta_j^2 = \mathrm{Var}(\log(T))$, where T is a random variable with density

$$r(t) = \frac{t^{y-\beta_j-1} e^{-t}}{\int_{A\tau_j}^{A\tau_{j+1}} s^{y-\beta_j-1} e^{-s} ds}, \qquad A\tau_j < t < A\tau_{j+1}.$$

Thus $\partial^2 h_{j,2}(y;\boldsymbol{\theta})/\partial\beta_j^2 \geq 0$ and $\partial h_{j,2}(y;\boldsymbol{\theta})/\partial\beta_j$ is increasing with respect to β_j. It follows then that $\partial h_{j,2}(y;\boldsymbol{\theta})/\partial\beta_j$ is bounded from above since h_j is bounded from above by Lemma C.1. Consequently, $\lim_{\beta_j\to\infty}\partial h_{j,2}(y;\boldsymbol{\theta})/\partial\beta_j$ exists. □

Assumption C.2 (Continuity Assumption)**.** *It is possible to extend the definition of $f(y;\boldsymbol{\theta})$ so that the range of $\boldsymbol{\theta}$ will be $\bar{\Theta}$ and so that, for any $\{\boldsymbol{\theta}_i\}$ and $\boldsymbol{\theta}^*$ in $\bar{\Theta}$, $\boldsymbol{\theta}_i \to \boldsymbol{\theta}^*$ implies*

$$f(y;\boldsymbol{\theta}) \to f(y;\boldsymbol{\theta}^*)$$

except perhaps on a set of y whose probability is 0 according to the probability density $f(y;\boldsymbol{\theta}_0)$. (The exceptional y-set may depend on $\boldsymbol{\theta}^*$ and $f(y;\boldsymbol{\theta}^*)$ need not be a probability density function.)

Proof. First, $f(y;\boldsymbol{\theta})$ is continuous with respect to $\boldsymbol{\theta} \in \Theta$ and thus f automatically fulfills the above continuity requirement for $\boldsymbol{\theta} \in \Theta$. Define $\partial\Theta = \bar{\Theta}\backslash\Theta$. Now, we will show that we can define $f(y;\boldsymbol{\theta}^*)$, where $\boldsymbol{\theta}^* \in \partial\Theta$, as $\lim_{\boldsymbol{\theta}\to\boldsymbol{\theta}^*} f(y;\boldsymbol{\theta})$. It is thus only required to show the existence of this limit. Notice that $\lim_{\boldsymbol{\theta}\to\boldsymbol{\theta}^*} f(y;\boldsymbol{\theta})$ exists for boundary points $\boldsymbol{\theta} \in \partial\Theta$ with $|\boldsymbol{\theta}|_E \neq \infty$. The remaining case $|\boldsymbol{\theta}|_E = \infty$ can be separated into three sub-cases: (i) $|\boldsymbol{\beta}|_E = \infty$ and $|\boldsymbol{\tau}|_E < \infty$, (ii) $|\boldsymbol{\beta}|_E < \infty$ and $|\boldsymbol{\tau}|_E = \infty$, and (iii) $|\boldsymbol{\beta}|_E = \infty$ and $|\boldsymbol{\tau}|_E = \infty$.

1. Suppose $|\boldsymbol{\beta}|_E = \infty$ and $|\boldsymbol{\tau}|_E < \infty$. From Lemma C.2, $\lim_{|\boldsymbol{\beta}|_E\to\infty} f(y;\boldsymbol{\theta})$ exists.
2. Suppose $|\boldsymbol{\beta}|_E < \infty$ and $|\boldsymbol{\tau}|_E = \infty$. This implies that there exists at least one j such that $\tau_j = \infty$. Here, we have

   $$0 \leq a^{\beta_j}\int_{Aa}^{Ab} t^{y-\beta_j-1}e^{-t}dt \leq a^{\beta_j}\int_{Aa}^{\infty} t^{y-\beta_j-1}e^{-t}dt,$$

   where $0 < a < b$. Taking the limit on the right-hand side, using the l'Hospital rule, it follows that

   $$\lim_{a\to\infty}\frac{\int_{Aa}^{\infty} t^{y-\beta_j-1}e^{-t}dt}{a^{-\beta_j}} = \lim_{a\to\infty}\frac{A^{y-\beta_j-1}\tau_j^y e^{-Aa}}{\beta_j} = 0.$$

   Since $0 \leq (c/a)^{\beta_{j-1}} \leq 1$ for all $0 < c < a$, $\lim_{|\boldsymbol{\tau}|_E\to\infty} f(y;\boldsymbol{\theta})$ exists.
3. $|\boldsymbol{\beta}|_E = \infty$ and $|\boldsymbol{\tau}|_E = \infty$. The existence of $\lim_{|\boldsymbol{\theta}|_E\to\infty} f(y;\boldsymbol{\theta})$ is basically implied by Lemma C.1.

The proof is complete. □

Assumption C.3. *For any $\boldsymbol{\theta}$ in $\bar{\Theta}$ and any $\rho > 0$, $w(y; \boldsymbol{\theta}, \rho)$ is a measurable function of y, where*

$$w(y; \boldsymbol{\theta}, \rho) = \sup f(y; \boldsymbol{\theta}'),$$

the supremum being taken over all $\boldsymbol{\theta}'$ in $\bar{\Theta}$ for which $\delta(\boldsymbol{\theta}, \boldsymbol{\theta}') < \rho$.

Proof. The statement is implied by the continuity of $f(y; \boldsymbol{\theta})$ with respect to $\boldsymbol{\theta} \in \bar{\Theta}$. □

Assumption C.4 (Identifiability Assumption)**.** *If $\boldsymbol{\theta}$ in $\bar{\Theta}$ is different from $\boldsymbol{\theta}_0$, then, for at least one x,*

$$\int_{-\infty}^{x} f(y|\boldsymbol{\theta}) d\mu \neq \int_{-\infty}^{x} f(y|\boldsymbol{\theta}_0) d\mu,$$

the integral being over those y all of whose components $\leq$ the corresponding of x.

Proof. In the present case, μ is the counting measure and thus, for all $\boldsymbol{\theta} \in \bar{\Theta}$, if $f(y|\boldsymbol{\theta}) \neq f(y|\boldsymbol{\theta}_0)$ for at least one $y \in \mathbb{N}_0$, it fulfills the above assumption. This is obviously true for $\boldsymbol{\theta} \in \Theta$. Since $\boldsymbol{\theta}_0 \in \Theta$, it is also easy to see that the above is true for $\boldsymbol{\theta} \in \bar{\Theta}$. □

Assumption C.5 (Integrability Assumption)**.** *For any $\boldsymbol{\theta}$ in $\bar{\Theta}$ we have*

$$\lim_{\rho \downarrow 0} \mathbb{E}\left[\log \frac{w(Y; \boldsymbol{\theta}, \rho)}{f(Y; \boldsymbol{\theta}_0)}\right]^{+} < \infty,$$

where w is defined in Assumption C.3.

Proof. Since $f(y; \boldsymbol{\theta})$ is continuous and bounded over $\bar{\Theta}$, $\log w(y; \boldsymbol{\theta}, \rho)$ is bounded from above. Now, we want to show that $\mathbb{E}|\log f(Y; \boldsymbol{\theta}_0)| < \infty$. Since $f(y; \boldsymbol{\theta}_0)$ is bounded from above, we only need $\mathbb{E}\{\log(f(Y; \boldsymbol{\theta}_0)\} > -\infty$, which can be shown as follows.

Note that, for any $\boldsymbol{\theta} \in \Theta$,

$$\begin{aligned}
\mathbb{E}[\log\{f(Y;\boldsymbol{\theta})\}] &= \mathbb{E}\left[\log\left\{\sum_{j=1}^{B}\left(\frac{\tau_{j-1}}{\tau_j}\right)^{\beta_{j-1}}\frac{\beta_j(A\tau_j)^{\beta_j}}{Y!}\int_{A\tau_j}^{A\tau_{j+1}} t^{Y-\beta_j-1}e^{-t}dt\right\}\right] \\
&\geq \sum_{j=1}^{B}\left[\log\left\{\left(\frac{\tau_{j-1}}{\tau_j}\right)^{\beta_{j-1}}\beta_j(A\tau_j)^{\beta_j}\right\} + \mathbb{E}\left\{\log\left(\frac{\int_{A\tau_j}^{A\tau_{j+1}} t^{Y-\beta_j-1}e^{-t}dt}{Y!}\right)\right\}\right] \\
&\geq \sum_{j=1}^{B}\log\left\{\left(\frac{\tau_{j-1}}{\tau_j}\right)^{\beta_{j-1}}\beta_j(A\tau_j)^{\beta_j}\right\} + \sum_{j=1}^{B}\mathbb{E}\left\{\log\left(\frac{\int_{A\tau_j}^{\infty} t^{Y-\beta_j-1}e^{-t}dt}{Y!}\right)\right\} \\
&= \sum_{j=1}^{B}\log\left\{\left(\frac{\tau_{j-1}}{\tau_j}\right)^{\beta_{j-1}}\beta_j(A\tau_j)^{\beta_j}\right\} + \sum_{j=1}^{B}\mathbb{E}\left[\log\left\{\frac{\Gamma(Y-\beta_j, A\tau_j)}{\Gamma(Y+1)}\right\}\right]
\end{aligned}$$

Here,

$$\frac{\Gamma(Y-\beta_j, A\tau_j)}{\Gamma(Y+1)} = \frac{\Gamma(Y-\beta_j, A\tau_j)}{\Gamma(Y-\beta_j)}\frac{\Gamma(Y-\beta_j)}{\Gamma(Y+1)} = Q(Y-\beta_j, A\tau_j)\frac{\Gamma(Y-\beta_j)}{\Gamma(Y+1)},$$

where Q is the regularized incomplete gamma function. Now, we state the asymptotic expansions of the regularized incomplete gamma function and the ratio of two gamma functions: When $a \to \infty$,

$$\begin{aligned}
Q(a,z) &\propto 1 - \frac{a^{-a-1/2}e^{a-z}z^a}{\sqrt{2\pi}}\left\{1 + O\left(\frac{1}{a}\right)\right\}, \\
\frac{\Gamma(a+b)}{\Gamma(a+c)} &\propto a^{b-c}\left\{1 + O\left(\frac{1}{a}\right)\right\}.
\end{aligned}$$

Applying these asymptotic expansions for large y,

$$\frac{\Gamma(y-\beta_j, A\tau_j)}{\Gamma(y+1)} \propto y^{-\beta_j-1}\left\{1 + O\left(\frac{1}{y}\right)\right\}.$$

Thus, in order to bound $\mathbb{E}[\log\{\Gamma(Y-\beta_j, A\tau_j)/\Gamma(Y+1)\}]$, we only have to bound $\mathbb{E}\{\log(Y)1_{\{Y\geq M\}}\}$ away from ∞ for sufficiently large M. Here, we define $\log 0 \times 0 = 0$. Now, we only have to consider the boundedness of $\sum_{y=M}^{\infty}\log(y)/y^{\beta_j+1}$ for $j = 1, \ldots, B$. It is bounded whenever $\beta_j > 0$, which is fulfilled by any $\boldsymbol{\theta} \in \Theta$. Thus, $\mathbb{E}\{\log(f(Y;\boldsymbol{\theta}_0)\} > -\infty$ since $\boldsymbol{\theta}_0 \in \Theta$. □

The statement of Theorem 5.1 follows now from Section 2 of Kiefer and Wolfowitz (1956).

References

Abramowitz, M. and Stegun, I. A. (1964) *Handbook of Mathematical Functions: With Formulars, Graphs, and Mathematical Tables*, vol. 55. National Bureau of Standards.

Arsigny, V., Fillard, P., Pennec, X. and Ayache, N. (2006) Log-euclidean metrics for fast and simple calculus on diffusion tensors. *Magnetic resonance in medicine*, **56**, 411–421.

Aue, A. and Lee, T. C. M. (2011) On image segmentation using information theoretic criteria. *The Annals of Statistics*, **39**, 2912–2935.

Baines, P., Udaltsova, I., Zezas, A. and Kashyap, V. (2012a) Bayesian estimation of log n — log s. In *Statistical Challenges in Modern Astronomy V* (eds. E. D. Feigelson and G. J. Babu), Lecture Notes in Statistics, 469–472. Springer New York. URL `http://dx.doi.org/10.1007/978-1-4614-3520-4_43`.

Baines, P. D. (2010) *Statistics, Science and Statistical Science: Modeling, Inference and Computation with Applications to the Physical Sciences*. Ph.D. thesis.

Baines, P. D., Meng, X.-L. and Xie, X. (2012b) The interwoven EM algorithm. *Submitted*.

Bammer, R., Holdsworth, S. J., Veldhuis, W. B. and Skare, S. T. (2009) New methods in diffusion-weighted and diffusion tensor imaging. *Magnetic resonance imaging clinics of North America*, **17**, 175–204.

Basser, P. J., Mattiello, J. and LeBihan, D. (1994) Mr diffusion tensor spectroscopy and imaging. *Biophysical journal*, **66**, 259–267.

Basser, P. J., Pajevic, S., Pierpaoli, C., Duda, J. and Aldroubi, A. (2000) In vivo fiber tractography using dt-mri data. *Magnetic Resonance in Medicine*, **44**, 625–632.

Beaulieu, C. (2002) The basis of anisotropic water diffusion in the nervous system – a technical review. *NMR in Biomedicine*, **15**, 435–455.

Behrens, T., Berg, H. J., Jbabdi, S., Rushworth, M. and Woolrich, M. (2007) Probabilistic diffusion tractography with multiple fibre orientations: What can we gain? *Neuroimage*, **34**, 144–155.

Behrens, T., Woolrich, M., Jenkinson, M., Johansen-Berg, H., Nunes, R., Clare, S., Matthews, P., Brady, J. and Smith, S. (2003) Characterization and propagation of uncertainty in diffusion-weighted mr imaging. *Magnetic Resonance in Medicine*, **50**, 1077–1088.

Bhattacharya, A. and Bhattacharya, R. (2012) *Nonparametric inference on manifolds: with applications to shape spaces*. Cambridge University Press.

Borwein, J. M. and Chan, O.-Y. (2009) Uniform bounds for the complementary incomplete gamma function. *Mathematical Inequalities & Applications*, **12**, 115–121.

Byrd, R. H., Lu, P., Nocedal, J. and Zhu, C. (1995) A limited memory algorithm for bound constrained optimization. *SIAM Journal on Scientific Computing*, **16**, 1190–1208.

Cappelluti, N., Hasinger, G., Brusa, M., Comastri, A., Zamorani, G., Böhringer, H., Brunner, H., Civano, F., Finoguenov, A., Fiore, F., Gilli, R., Griffiths, R. E., Mainieri, V., Matute, I., Miyaji, T. and Silverman, J. (2007) The XMM-Newton Wide-Field Survey in the COSMOS Field. II. X-ray data and the log N–log S relations. *The Astrophysical Journal Supplement Series*, **172**, 341.

Carmichael, O., Chen, J., Paul, D. and Peng, J. (2013) Diffusion tensor smoothing through weighted karcher means. *Electronic Journal of Statistics*, **7**, 1913–1956.

Chanraud, S., Zahr, N., Sullivan, E. V. and Pfefferbaum, A. (2010) Mr diffusion tensor imaging: a window into white matter integrity of the working brain. *Neuropsychology review*, **20**, 209–225.

Davison, A. C. (1997) *Bootstrap Methods and their Application*. New York: Cambridge University Press.

Dempster, A. P., Laird, N. M. and Rubin, D. B. (1977) Maximum likelihood from incomplete data via the EM algorithm. *Journal of the Royal Statistical Society, Series B*, **39**, 1–38.

Descoteaux, M., Angelino, E., Fitzgibbons, S. and Deriche, R. (2007) Regularized, fast, and robust analytical q-ball imaging. *Magnetic Resonance in Medicine*, **58**, 497–510.

Efron, B. and Tibshirani, R. J. (1994) *An introduction to the bootstrap*, vol. 57. CRC press.

Fan, J. and Gijbels, I. (1996) *Local Polynomial Modelling and Its Applications*. London: Chapman and Hall.

Fillard, P., Pennec, X., Arsigny, V. and Ayache, N. (2007) Clinical dt-mri estimation, smoothing, and fiber tracking with log-euclidean metrics. *Medical Imaging, IEEE Transactions on*, **26**, 1472–1482.

Fletcher, P. T. and Joshi, S. (2007) Riemannian geometry for the statistical analysis of diffusion tensor data. *Signal Processing*, **87**, 250–262.

Friel, N. and Pettitt, A. (2008) Marginal likelihood estimation via power posteriors. *Journal of the Royal Statistical Society: Series B*, **70**, 589–607.

Friman, O., Farneback, G. and Westin, C.-F. (2006) A bayesian approach for stochastic white matter tractography. *Medical Imaging, IEEE Transactions on*, **25**, 965–978.

Ginsbourger, D., Riche, R. and Carraro, L. (2010) Kriging is well-suited to parallelize optimization. *Computational intelligence in expensive optimization problems*, 131–162.

Gu, C. (2013) *Smoothing spline ANOVA models*, vol. 297. Springer, second edition edn.

Gudbjartsson, H. and Patz, S. (1995) The rician distribution of noisy mri data. *Magnetic Resonance in Medicine*, **34**, 910–914.

Guetta, D., Granot, J. and Begelman, M. C. (2005) Constraining the structure of gamma-ray burst jets through the log N–log S distribution. *The Astrophysical Journal*, **622**, 482–491.

Hall, P. (1992a) Effect of bias estimation on coverage accuracy of bootstrap confidence intervals for a probability density. *The Annals of Statistics*, **20**, 675–694.

Hall, P. (1992b) On bootstrap confidence intervals in nonparametric regression. *The Annals of Statistics*, **20**, 695–711.

Härdle, W. and Bowman, A. W. (1988) Bootstrapping in nonparametric regression: Local adaptive smoothing and confidence bands. *Journal of the American Statistical Association*, **83**, 102–110.

Hewish, A. (1961) Extrapolation of the number-flux density relation of radio stars by Scheuer's statistical methods. *Monthly Notices of the Royal Astonomical Society*, **123**, 167.

Hickox, R. C. and Markevitch, M. (2007) Can Chandra resolve the remaining cosmic X-ray background? *The Astrophysical Journal*, **671**, 1523–1530.

Higdon, D., Kennedy, M., Cavendish, J. C., Cafeo, J. A. and Ryne, R. D. (2004) Combining field data and computer simulations for calibration and prediction. *SIAM Journal on Scientific Computing*, **26**, 448–466.

Hosey, T., Williams, G. and Ansorge, R. (2005) Inference of multiple fiber orientations in high angular resolution diffusion imaging. *Magnetic Resonance in Medicine*, **54**, 1480–1489.

Huang, D., Allen, T. T., Notz, W. I. and Miller, R. A. (2006a) Sequential kriging optimization using multiple-fidelity evaluations. *Structural and Multidisciplinary Optimization*, **32**, 369–382.

Huang, D., Allen, T. T., Notz, W. I. and Zeng, N. (2006b) Global optimization of stochastic black-box systems via sequential kriging meta-models. *Journal of Global Optimization*, **34**, 441–466.

Jones, D. R., Schonlau, M. and Welch, W. J. (1998) Efficient global optimization of expensive black-box functions. *Journal of Global optimization*, **13**, 455–492.

Jordán, A., Côté, P., Ferrarese, L., Blakeslee, J. P., Mei, S., Merritt, D., Milosavljevifá, M., Peng, E. W., Tonry, J. L. and West, M. J. (2004) The ACS Virgo Cluster Survey. III. Chandra and Hubble space telescope observations of low-mass X-ray binaries and globular clusters in M87. *The Astrophysical Journal*, **613**, 279.

Kaufman, L. and Rousseeuw, P. J. (1990) *Finding groups in data: an introduction to cluster analysis*, vol. 344. New Jersey: John Wiley & Sons.

Kennedy, M. C. and O'Hagan, A. (2001) Bayesian calibration of computer models. *Journal of the Royal Statistical Society: Series B*, **63**, 425–464.

Kenter, A. T. and Murray, S. S. (2003) A new technique for determining the number of X-ray sources per flux density interval. *The Astrophysical Journal*, **584**, 1016–1020.

Kiefer, J. and Wolfowitz, J. (1956) Consistency of the maximum likelihood estimator in the presence of infinitely many incidental parameters. *The Annals of Mathematical Statistics*, **27**, 887–906.

Kitayama, T., Sasaki, S. and Suto, Y. (1998) Cosmological implications of number counts of clusters of galaxies: log N–log S in X-ray and submm bands. *Publications of the Astronomical Society of Japan*, **50**, 1–11.

Koch, M. A., Norris, D. G. and Hund-Georgiadis, M. (2002) An investigation of functional and anatomical connectivity using magnetic resonance imaging. *Neuroimage*, **16**, 241–250.

Kouzu, T., Tashiro, M. S., Terada, Y., Yamada, S., Bamba, A., Enoto, T., Mori, K., Fukazawa, Y. and Makishima, K. (2013) Spectral Variation of Hard X-Ray Emission from the Crab Nebula with the Suzaku Hard X-Ray Detector. *Publications of the Astronomical Society of Japan*, **65**, 74.

Lin, Y. and Zhang, H. H. (2006) Component selection and smoothing in smoothing spline analysis of variance models. *Annals of Statistics*, **34**, 2272–2297.

Loeppky, J. L., Moore, L. M. and Williams, B. J. (2010) Batch sequential designs for computer experiments. *Journal of Statistical Planning and Inference*, **140**, 1452–1464.

Mammen, E. (1991) Nonparametric regression under qualitative smoothness assumptions. *The Annals of Statistics*, **19**, 741–759.

Mateos, S., Warwick, R. S., Carrera, F. J., Stewart, G. C., Ebrero, J., Della Ceca, R., Caccianiga, A., Gilli, R., Page, M. J., Treister, E., Tedds, J. A., Watson, M. G., Lamer, G., Saxton, R. D., Brunner, H. and Page, C. G. (2008) High precision X-ray log N–log S distributions: Implications for the obscured AGN population. *Astronomy & Astrophysics*, **492**, 51–69.

Mathiesen, B. and Evrard, A. E. (1998) Constraints on Ω_0 and cluster evolution using the ROSAT log N–log S relation. *Monthly Notices of the Royal Astronomical Society*, **295**, 769–780.

Minh, H. Q. (2010) Some properties of gaussian reproducing kernel hilbert spaces and their implications for function approximation and learning theory. *Constructive Approximation*, **32**, 307–338.

Moretti, A., Campana, S., Lazzati, D. and Tagliaferri, G. (2003) The resolved fraction of the cosmic X-ray background. *The Astrophysical Journal*, **588**, 696–703.

Mori, S. (2007) *Introduction to diffusion tensor imaging*. Amsterdam: Elsevier.

Mori, S., Crain, B. J., Chacko, V. and Van Zijl, P. (1999) Three-dimensional tracking of axonal projections in the brain by magnetic resonance imaging. *Annals of neurology*, **45**, 265–269.

Mori, S. and van Zijl, P. (2002) Fiber tracking: principles and strategies–a technical review. *NMR in Biomedicine*, **15**, 468–480.

Mukherjee, P., Berman, J., Chung, S., Hess, C. and Henry, R. (2008) Diffusion tensor mr imaging and fiber tractography: theoretic underpinnings. *American journal of neuroradiology*, **29**, 632–641.

Parker, G. J. M. and Alexander, D. C. (2003) Probabilistic monte carlo based mapping of cerebral connections utilising whole-brain crossing fibre information. In *Information Processing in Medical Imaging*, 684–695. Springer.

Pennec, X., Fillard, P. and Ayache, N. (2006) A riemannian framework for tensor computing. *International Journal of Computer Vision*, **66**, 41–66.

Reich, B. J., Storlie, C. B. and Bondell, H. D. (2009) Variable selection in bayesian smoothing spline anova models: Application to deterministic computer codes. *Technometrics*, **51**.

Rousseeuw, P. J. (1987) Silhouettes: a graphical aid to the interpretation and validation of cluster analysis. *Journal of computational and applied mathematics*, **20**, 53–65.

Ruppert, D., Wand, M. P. and Carroll, R. J. (2003) *Semiparametric regression*. No. 12. Cambridge University Press.

Ryde, F. (1999) Smoothly Broken Power Law Spectra of Gamma-Ray Bursts. *Astrophysical Letters and Communications*, **39**, 281.

Scherrer, B. and Warfield, S. K. (2010) Why multiple b-values are required for multi-tensor models. evaluation with a constrained log-euclidean model. In *2010 IEEE International Symposium on Biomedical Imaging: From Nano to Macro*, 1389–1392.

Scheuer, P. A. G. (1957) A statistical method for analysing observations of faint radio stars. *Proceedings of the Cambridge Philosophical Society*, **53**, 764–773.

Schwarz, G. (1978) Estimating the dimension of a model. *The Annals of Statistics*, **6**, 461–464.

Segura, C., Lazzati, D. and Sankarasubramanian, A. (2013) The use of broken power-laws to describe the distributions of daily flow above the mean annual flow across the conterminous U.S. *Journal of Hydrology*, **505**, 35–46.

Sekhon, J. S. and Mebane, W. R. (1998) Genetic optimization using derivatives. *Political Analysis*, **7**, 187–210.

Sporns, O. (2011) *Networks of the Brain*. The MIT Press.

Steinwart, I., Hush, D. and Scovel, C. (2006) An explicit description of the reproducing kernel hilbert spaces of gaussian rbf kernels. *IEEE Transactions on Information Theory*, **52**, 4635–4643.

Storlie, C. B., Bondell, H. D., Reich, B. J. and Zhang, H. H. (2011) Surface estimation, variable selection, and the nonparametric oracle property. *Statistica Sinica*, **21**, 679.

Storlie, C. B. and Helton, J. C. (2008) Multiple predictor smoothing methods for sensitivity analysis: Description of techniques. *Reliability Engineering & System Safety*, **93**, 28–54.

Storlie, C. B., Lane, W. A., Ryan, E. M., Gattiker, J. R. and Higdon, D. M. (2013a) Calibration of computational models with categorical parameters and correlated outputs via bayesian smoothing spline anova. Journal of the American Statistical Association (in review).

Storlie, C. B., Lane, W. A., Ryan, E. M., Gattiker, J. R. and Higdon, D. M. (2014) Calibration of computational models with categorical parameters and correlated outputs via bayesian smoothing spline anova. Submitted.

Storlie, C. B., Reich, B. J., Helton, J. C., Swiler, L. P. and Sallaberry, C. J. (2013b) Analysis of computationally demanding models with continuous and categorical inputs. *Reliability Engineering & System Safety*, **113**, 30–41.

Storlie, C. B., Swiler, L. P., Helton, J. C. and Sallaberry, C. J. (2009) Implementation and evaluation of nonparametric regression procedures for sensitivity analysis of computationally demanding models. *Reliability Engineering & System Safety*, **94**, 1735–1763.

Tabelow, K., Voss, H. and Polzehl, J. (2012) Modeling the orientation distribution function by mixtures of angular central gaussian distributions. *Journal of neuroscience methods*, **203**, 200–211.

Tibshirani, R. (1996) Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society: Series B*, **58**, 267–288.

Tournier, J., Calamante, F., Connelly, A. *et al.* (2007) Robust determination of the fibre orientation distribution in diffusion mri: non-negativity constrained super-resolved spherical deconvolution. *NeuroImage*, **35**, 1459–1472.

Tournier, J., Calamante, F., Gadian, D. G., Connelly, A. *et al.* (2004) Direct estimation of the fiber orientation density function from diffusion-weighted mri data using spherical deconvolution. *NeuroImage*, **23**, 1176–1185.

Trudolyubov, S. P., Borozdin, K. N., Priedhorsky, W. C., Mason, K. O. and Cordova, F. A. (2002) On the X-ray source luminosity distributions in the bulge and disk of M31: First results from the XMM-Newton Survey. *The Astrophysical Journal Letters*, **571**, 17–21.

Tuch, D. S. (2002) *Diffusion MRI of complex tissue structure*. Ph.D. thesis, Massachusetts Institute of Technology.

Tuch, D. S. (2004) Q-ball imaging. *Magnetic Resonance in Medicine*, **52**, 1358–1372.

Tuch, D. S., Reese, T. G., Wiegell, M. R., Makris, N., Belliveau, J. W. and Wedeen, V. J. (2002) High angular resolution diffusion imaging reveals intravoxel white matter fiber heterogeneity. *Magnetic Resonance in Medicine*, **48**, 577–582.

Van De Geer, S. (2000) *Empirical Processes in M-estimation*. New York: Cambridge University Press.

Van der Vaart, A. W. (2000) *Asymptotic Statistics*. New York: Cambridge University Press.

Vecchia, A. V. and Cooley, R. L. (1987) Simultaneous confidence and prediction intervals for nonlinear regression models with application to a groundwater flow model. *Water Resources Research*, **23**, 1237–1250.

Wahba, G. (1990) *Spline Models for Observational Data*, vol. 59. Philadelphia: SIAM.

Wald, A. (1949) Note on the consistency of the maximum likelihood estimate. *The Annals of Mathematical Statistics*, **20**, 595–601.

Weinstein, D., Kindlmann, G. and Lundberg, E. (1999) Tensorlines: Advection-diffusion based propagation through diffusion tensor fields. In *Proceedings of the conference on Visualization*, 249–253.

Wiegell, M. R., Larsson, H. B. and Wedeen, V. J. (2000) Fiber crossing in human brain depicted with diffusion tensor mr imaging1. *Radiology*, **217**, 897–903.

Wong, R. K. W., Baines, P., Aue, A., Lee, T. C. M. and Kashyap, V. L. (2014) Automatic estimation of flux distributions of astrophysical source populations. *Annals of Applied Statistics*, to appear.

Yao, Y.-C. (1988) Estimating the number of change-points via Schwarz' criterion. *Statistics & Probability Letters*, **6**, 181–189.

Yu, Y. and Meng, X.-L. (2011) To center or not to center: That is not the question — An ancillarity-sufficiency interweaving strategy (ASIS) for boosting MCMC efficiency. *Journal of Computational and Graphical Statistics*, **20**, 531–570.

Yuan, Y., Zhu, H., Lin, W. and Marron, J. S. (2012) Local polynomial regression for symmetric positive definite matrices. *Journal of the Royal Statistical Society: Series B (Statistical Methodology)*, **74**, 697–719.

Zou, H. (2006) The adaptive lasso and its oracle properties. *Journal of the American Statistical Association*, **101**, 1418–1429.