9-2010

A Hierarchical Spherical Radial Quadrature Algorithm for Multilevel GLMMS, GSMMS, and Gene Pathway Analysis

Jacob A. Gagnon
University of Massachusetts Amherst, jakegagnon@yahoo.com

A HIERARCHICAL SPHERICAL RADIAL QUADRATURE ALGORITHM FOR MULTILEVEL GLMMS, GSMMS, AND GENE PATHWAY ANALYSIS

A Dissertation Presented

by

JACOB A. GAGNON

Submitted to the Graduate School of the
University of Massachusetts Amherst in partial fulfillment
of the requirements for the degree of

DOCTOR OF PHILOSOPHY

September 2010

Department of Mathematics and Statistics

A HIERARCHICAL SPHERICAL RADIAL QUADRATURE ALGORITHM FOR MULTILEVEL GLMMS, GSMMS, AND GENE PATHWAY ANALYSIS

A Dissertation Presented

by

JACOB A. GAGNON

Approved as to style and content by:

Anna Liu, Chair

Daeyoung Kim, Member

John Staudenmayer, Member

Rongheng Lin, Member

George Avrunin, Department Head
Mathematics and Statistics

ACKNOWLEDGMENTS

I am very thankful towards my thesis adviser, Professor Anna Liu, of the University of Massachusetts Amherst. She guided my professional development as a statistician by offering constructive advice on all aspects of my dissertation: the manuscript preparation, the statistical theory, and the computational algorithms. This thesis would not have been possible without her advice and support. I am looking forward to our future collaboration together. I would also like to thank my thesis committee for their suggestions on how to improve my dissertation. Lastly, I would like to thank my friends and family for their support throughout this research.

ABSTRACT

A HIERARCHICAL SPHERICAL RADIAL QUADRATURE ALGORITHM FOR MULTILEVEL GLMMs, GSMMs, and GENE PATHWAY ANALYSIS
SEPTEMBER 2010
JACOB A. GAGNON,
B.S., MASSACHUSETTS INSTITUTE OF TECHNOLOGY
M.S., UNIVERSITY OF MASSACHUSETTS AT AMHERST
Ph.D., UNIVERSITY OF MASSACHUSETTS AT AMHERST
Directed by: Professor Anna Liu

The first part of my thesis is concerned with estimation for longitudinal data using generalized semi-parametric mixed models and multilevel generalized linear mixed models for a binary response. Likelihood based inferences are hindered by the lack of a closed form representation. Consequently, various integration approaches have been proposed. We propose a spherical radial integration based approach that takes advantage of the hierarchical structure of the data, which we call the 2 SR method. Compared to Pinheiro and Chao's multilevel Adaptive Gaussian quadrature [37], our proposed method has an improved time complexity with the number of functional evaluations scaling linearly in the number of subjects and in the dimension of random effects per level. Simulation studies show that our approach has similar to better accuracy compared to Gauss Hermite Quadrature (GHQ) and has better accuracy compared to PQL especially in the variance components.

The second part of my thesis is concerned with identifying differentially expressed gene pathways/gene sets. We propose a logistic kernel machine to model the gene pathway effect with a binary response. Kernel machines were chosen since they account for gene interactions and clinical covariates. Furthermore, we established a connection between our logistic kernel machine with GLMMs allowing us to use ideas from the GLMM literature. For estimation and testing, we adopted Clarkson's spherical radial approach [6] to perform the high dimensional integrations. For estimation, our performance in simulation studies is comparable to better than Bayesian approaches at a much lower computational cost. As for testing of the genetic pathway effect, our REML likelihood ratio test has increased power compared to a score test for simulated non-linear pathways. Additionally, our approach has three main advantages over previous methodologies: 1) our testing approach is self-contained rather than competitive, 2) our kernel machine approach can model complex pathway effects and gene-gene interactions, and 3) we test for the pathway effect adjusting for clinical covariates. Motivation for our work is the analysis of an Acute Lymphocytic Leukemia data set where we test for the genetic pathway effect and provide confidence intervals for the fixed effects.

TABLE OF CONTENTS

LIST OF TABLES

CHAPTER 1

MULTILEVEL ANALYSIS

1.1 Introduction

Longitudinal and cluster data appear in a wide variety of disciplines: biology, epidemiology, and others. Typically mixed effect models are used to analyze these types of data to account for correlations within clusters. However, the routine use of the generalized mixed effects models (GLMM) has been hindered by the lack of closed-form likelihood functions. This led to various integration techniques being introduced in the literature. Unfortunately, for the case of a binary response with multilevel random effects, existing approximations remain unsatisfactory in terms of either computational cost or approximation accuracy, and improvements are needed. In this thesis, we introduce new algorithms for multilevel GLMMs with nested random effects that are based on the spherical-radial integration approximation by Monahan and Genz [34]. These algorithms greatly reduce the computational complexity compared to product quadrature and MCMC based ones, and the algorithms are more accurate than the fast Laplace based ones.

Integration approximations for GLMMs include product quadrature as defined by the product of one-dimensional quadratures, implemented in SAS NLMIXED, GLLAMM in STATA, MIXOR (Hedeker and Gibbons [19]), and MIXNO (Hedeker

[18]); First order PQL approximation in R's *glmmPQL* function and higher order Laplace approximations in HLM (Raudenbush, et al. [38]; Noh and Lee [36]); Spherical-radial integral approximation (Monahan and Genz, [34]; Clarkson and Zhan [6]); Monte Carlo EM (Booth and Hobert [1]); and MCMC in a Bayesian framework (Crainiceanu, Ruppert and Wand [7]; Chib and Jeliazkov [4]). Product quadrature is usually not a viable solution for high dimensional integration as the number of integrand evaluations needed increases exponentially with the dimension. Laplace and PQL approximations are fast, but they are essentially one-point quadrature, and they give biased estimates for binary data (Rodriguez and Goldman [39], Breslow and Clayton [5], and Pinheiro and Chao [37]). Higher-order Laplace and PQL approximations are better at reducing bias, but they are hard to implement due to the high order derivatives of the integrand (Rodriguez and Goldman [40]; Raudenbush [38]). Kuk's [21] parametric bootstrap method can correct the bias of these PQL methods but the method is computationally intensive (Rodriguez and Goldman [40]). Breslow and Lin [2] also proposed a bias correction method. Their method uses a 4th order Laplace approximation to correct the PQL bias. Improved estimation performance was observed for large variance components.

Other approaches to handle GLMMs include MCMC, spherical radial approximation [6], the marginal quasi-likelihood (MQL) methods of Breslow and Clayton [5], Longford's approximate likelihood [25] [26], Goldstein's generalized least squares (GLS) [15], and Goldstein's Quadratic Approximation [15]. The MCMC algorithm can be fine tuned to achieve any accuracy but it is computationally expensive. Clarkson and Zhan's spherical radial algorithm [6] had favorable performance compared to the second-order Laplace method and compared to the product quadrature method for one-level GLMMs. Longford's approximate likelihood,

Goldstein's generalized least squares (GLS), and MQL were shown to be equivalent by Rodriquez et al [39], so they have the same drawbacks as MQL, ie biased estimates for binary data (Rodriguez and Goldman [39]; Breslow and Clayton [5]; Pinheiro [37]). Goldstein's quadratic approximation improves upon his GLS approach, but his fixed and random effect estimates still have a downward bias. In this thesis, we extend the spherical-radial algorithm to GLMMs with multiple levels of random effects. As expected, it improves the PQL based estimates, especially those of the variance components, with affordable computational cost.

For multilevel GLMMs, the problem of integration approximation becomes more prominent because the integration dimension depends on the total number of random effects which is linear in the number of subjects. All the algorithms developed for the one-level GLMM can be applied blindly per level to the multilevel GLMM case. However, due to the potentially large integration dimension, they either lose accuracy or computational speed. Gibbons [12] and Pinheiro and Chao [37] noticed that if the data has a hierarchical data structure, then integration can be performed for each level of the hierarchy rather than integrating the whole integral. Pinheiro and Chao [37] developed an algorithm that greatly reduces the computational complexity and memory usage compared to a direct application of a single-level AGQ (Adaptive Gaussian Quadrature) approximation algorithm to the multilevel case, but the reduced computational complexity is still high. For two-level GLMM, it is $nsub * N_{quad,outer}^{q_1} * N_{quad,inner}^{q_2}$ evaluations of the inner integrand and $N_{quad,outer}^{q_1}$ evaluations of the outer function where $N_{quad,outer}$ and $N_{quad,inner}$ represent the number of quadrature points for the outer and inner integrals respectively, $nsub$ represents the number of subjects, q_1 is the number of level one random effects, and q_2 is the number of level two random effects. Our aim is to create an algorithm which has the number of functional evaluations scaling linearly in the number of subjects and

in the number of random effects per level. To achieve this aim, we use two ideas: spherical transformation that preserves the hierarchical data structure and spherical radial integration. We choose to use spherical radial integration, because it has been shown to have less bias than the Laplace approximation and is more computational efficient than other approaches. Instead of applying the spherical radial algorithm once, we propose to apply it to each layer of the hierarchy to obtain a more accurate and stable integration approximation. Our algorithm's complexity for the two-level GLMM case is $N_{quad,outer} * (q_1 + 1)$ outer function evaluations and $(q_2+1) * N_{quad,inner} * nsub * N_{quad,outer} * (q_1+1)$ inner integrand function evaluations. Notice that the number of function evaluations is linear in the number of subjects, level one effects, and level two effects.

Another contribution of this thesis lies in estimation for generalized semiparametric mixed models (GSMM). GSMMs have been proposed and studied in Lin and Zhang [28], Karcher and Wang [20], Ruppert, Wand and Carroll [41], Gu and Ma [16], Chib and Jeliazkov [4], Zhao et al. [48], and Liang [22]. We note that when finite basis expansion is used to approximate the nonparametric function, for example, by a truncated power series, the GSMM becomes a multilevel GLMM, to which our algorithms can be applied. The connection between smoothing and mixed models was recognized and utilized by many authors (see Wand [45] and reference therein). For the kind of GSMM-equivalent GLMM we consider, the number of random effects is proportional to the number of spline knots, which could be large to allow flexible spline fitting. Therefore, a fast and accurate integration technique is even more critical in GSMM with splines.

This chapter is organized as follows. In section 1.2, we describe the spherical radial integration approximation technique. Then, in section 1.3, we derive our estimation procedure for a general two level GLMM. The algorithms presented in

this thesis generalize straightforwardly to the multilevel GLMM case. Two approaches to performing spherical radial integration are introduced in section 1.4. Connections between GSMMs and two-level GLMMs are discussed in section 1.5. Following this, a simulation study is performed in 1.6 to compare the estimation performance of the PQL method with the maximum likelihood method for GLMMs and GSMMs. In section 1.7, we present our HSR package which implements our methods. Lastly, in section 1.8, we discuss some future directions for our work.

1.2 Spherical radial algorithm

Let $f(\mathbf{x})$ be a n-dimensional function to be integrated. The spherical radial algorithm proposed by Monahan and Genz [34] begins by transforming the integrand, $f(\mathbf{x})$, into an approximately spherically symmetric integrand by using the integrand's mode and Hessian. The integrand's mode is given by $\hat{x} = argmax(ln(f(\mathbf{x})))$ and its Hessian is:

$$\mathbf{H} = -\frac{\partial^2 ln(f(\mathbf{x}))}{\partial \mathbf{x}^2}\bigg|_{\mathbf{x}=\hat{\mathbf{x}}} \tag{1.1}$$

Next, the Hessian is decomposed into its Cholesky's decomposition, so $\mathbf{H} = \mathbf{A}^T\mathbf{A}$. Using this decomposition, we can transform $f(\mathbf{x})$ into an approximately spherically symmetric function as follows:

$$f^*(\mathbf{a}) = |\mathbf{A}^{-1}| f(\hat{\mathbf{x}} + \mathbf{A}^{-1}\mathbf{a}) \tag{1.2}$$

Furthermore, let $\mathbf{a} = r\mathbf{z}$ where r is the radius and $\mathbf{z}$ is a vector on the unit sphere. Our integration then becomes:

$$\int f(\mathbf{x})dx = \int_0^\infty \int f^*(r\mathbf{z})d\mathbf{z}r^{n-1}dr \tag{1.3}$$

The outer integral over r can be approximated by a Gauss-Kronrod rule such as the 15-point or 31-point rule. However, numerical integration over r requires a

finite range of integration so we use a range of $(0, 4\sqrt{n-1})$ where $n \geq 2$. As for approximating the inner integral, Monahan and Genz [34] proposes a randomized integration rule of the form:

$$\int f^*(r\mathbf{z})d\mathbf{z} \approx \frac{1}{NRR_r} \sum_{i=1}^{NRR_r} \sum_{j=1}^{n+1} w_j f^*(r\mathbf{Q}_i\mathbf{v}_j) \tag{1.4}$$

The antipodal rule, the simplex rule, and the extended simplex rule are proposed by Monahan and Genz [34] as three ways to choose the weights, w_j, and nodal locations, $\mathbf{v}_j$. Here we use the simplex rule, where the nodes, $\mathbf{v}_j$, are a set of $n+1$ points symmetrically spaced on a n-dimensional sphere's surface. These nodes are also known as the standard simplex nodes. Monahan and Genz [34] defined the points, v_{jk}, on an unrotated simplex of radius 1 as:

$$\begin{aligned} v_{jk} &= 0, 0 < k < j < n+1 \\ v_{jj} &= \sqrt{\frac{(n+1)(n-j+1)}{n(n-j+2)}}, j = 1, \ldots, n \\ v_{jk} &= -\sqrt{\frac{n+1}{(n-j+1)n(n-j+2)}}, 0 < j < k \leq n+1 \end{aligned} \tag{1.5}$$

$\mathbf{Q}_i$ is an orthonormal random rotation matrix which acts on $\mathbf{v}_j$ to obtain a random rotation of the simplex points. Lastly, we define the weights, w_j, as $w_j = \frac{S_n}{n+1}$ where S_n is the surface area of an unit n-dimensional sphere and NRR_r is defined as the number of random rotations of the standard simplex at radius r.

1.3 Two-level GLMM

In this section, we describe the estimation procedure for a general two-level generalized linear mixed model. Let us assume that we have two levels of random effects: $\mathbf{b}_i^{(1)}$ and $\mathbf{b}_i^{(2)} = (\mathbf{b}_{i,1}^{(2)} \ldots \mathbf{b}_{i,nsub}^{(2)})$ where i is the group index, j is the subject

index, and *nsub* is the number of subjects. The negative log conditional density of group i for a binary response is then

$$p(\mathbf{y}_i|\mathbf{b}_i^{(1)}, \mathbf{b}_i^{(2)}) = -\sum_{j=1}^{nsub} \mathbf{y}_{ij} \cdot \boldsymbol{\eta}_{ij} + \sum_{j=1}^{nsub} \sum_{k=1}^{nobs} \log(1 + \exp(\eta_{ijk})) \tag{1.6}$$

where

$$\boldsymbol{\eta}_{ij} = logit(\boldsymbol{\mu}_{ij}) = \mathbf{X}_{ij}\boldsymbol{\beta} + \mathbf{A}_{ij}^{(1)}\mathbf{b}_i^{(1)} + \mathbf{A}_{ij}^{(2)}\mathbf{b}_{ij}^{(2)} \tag{1.7}$$

and k is the observation index. Let $\boldsymbol{\beta}$ be a vector of fixed effects and $\mathbf{X}_{i,j}$ be a matrix of fixed effect covariates. $\mathbf{A}_{ij}^{(1)}$ and $\mathbf{A}_{ij}^{(2)}$ are two sets of random effect covariates. We assume that our level one random effects, $\mathbf{b}_i^{(1)}$, are independent with density $N(0, \boldsymbol{\Sigma}^{(1)}(\boldsymbol{\phi}^{(1)}))$ and that our level two random effects, $\mathbf{b}_{ij}^{(2)}$, are independent with density $N(0, \boldsymbol{\Sigma}^{(2)}(\boldsymbol{\phi}^{(2)}))$. Furthermore, we let our level two effects be independent of the level one random effects. Estimates of the fixed effects, $\boldsymbol{\beta}$, can be obtained by minimizing:

$$\sum_{i=1}^{I} \sum_{j=1}^{nsub} \left[-\mathbf{y}_{ij} \cdot \boldsymbol{\eta}_{ij} + \sum_{k=1}^{nobs} \log(1 + \exp(\eta_{ijk})) \right] + \tag{1.8}$$
$$\sum_{i=1}^{I} \sum_{j=1}^{nsub} \left[\frac{1}{2} (\mathbf{b}_{ij}^{(2)})^T (\boldsymbol{\Sigma}^{(2)}(\boldsymbol{\phi}^{(2)}))^{(-1)} \mathbf{b}_{ij}^{(2)} \right] + \sum_{i=1}^{I} \frac{1}{2} (\mathbf{b}_i^{(1)})^T (\boldsymbol{\Sigma}^{(1)}(\boldsymbol{\phi}^{(1)}))^{-1} \mathbf{b}_i^{(1)}$$

where I is the number of groups. Using the negative log joint distribution, $p(\mathbf{y}_{ij}, \mathbf{b}_{ij}^{(2)}|\boldsymbol{\beta}, \mathbf{b}_i^{(1)})$, we can define ML estimates of $(\boldsymbol{\beta}, \boldsymbol{\phi}^{(1)}, \boldsymbol{\phi}^{(2)})$ as the maximizer of:

$$\prod_{i=1}^{I} \int \exp \left[\sum_{j=1}^{nsub} -p(\mathbf{y}_{ij}, \mathbf{b}_{ij}^{(2)}|\boldsymbol{\beta}, \mathbf{b}_i^{(1)}) \right] \exp \left[-\frac{1}{2} (\mathbf{b}_i^{(1)})^T (\boldsymbol{\Sigma}^{(1)}(\boldsymbol{\phi}^{(1)}))^{(-1)} \mathbf{b}_i^{(1)} \right] d\mathbf{b}_i^{(2)} d\mathbf{b}_i^{(1)} \tag{1.9}$$

where $p(\mathbf{y}_{ij}, \mathbf{b}_{ij}^{(2)}|\boldsymbol{\beta}, \mathbf{b}_i^{(1)})$ equals:

$$-\mathbf{y}_{ij} \cdot \boldsymbol{\eta}_{ij} + \sum_{k=1}^{nobs} \log(1 + \exp(\eta_{ijk})) + \frac{1}{2} (\mathbf{b}_{ij}^{(2)})^T (\boldsymbol{\Sigma}^{(2)}(\boldsymbol{\phi}^{(2)}))^{(-1)} \mathbf{b}_{ij}^{(2)} \tag{1.10}$$

Plugging eq. 1.10 into eq. 1.9 yields:

$$\prod_{i=1}^{I}\int \exp\left[\sum_{j=1}^{nsub}\mathbf{y}_{ij}\cdot\boldsymbol{\eta}_{ij}-\sum_{j=1}^{nsub}\sum_{k=1}^{nobs}\log(1+\exp(\eta_{ijk}))-\sum_{j=1}^{nsub}\frac{1}{2}(\mathbf{b}_{ij}^{(2)})^T(\boldsymbol{\Sigma}^{(2)}(\boldsymbol{\phi}^{(2)}))^{(-1)}\mathbf{b}_{ij}^{(2)}\right]*$$
$$\exp\left[-\frac{1}{2}(\mathbf{b}_i^{(1)})^T(\boldsymbol{\Sigma}^{(1)}(\boldsymbol{\phi}^{(1)}))^{(-1)}\mathbf{b}_i^{(1)}\right]d\mathbf{b}_i^{(2)}d\mathbf{b}_i^{(1)} \tag{1.11}$$

REML estimates of $(\boldsymbol{\phi}^{(1)},\boldsymbol{\phi}^{(2)})$ can be obtained by maximizing:

$$\prod_{i=1}^{I}\int \exp\left[\sum_{j=1}^{nsub}-p(\mathbf{y}_{ij},\mathbf{b}_{ij}^{(2)}|\boldsymbol{\beta},\mathbf{b}_i^{(1)})\right]\exp\left[-\frac{1}{2}(\mathbf{b}_i^{(1)})^T(\boldsymbol{\Sigma}^{(1)}(\boldsymbol{\phi}^{(1)}))^{(-1)}\mathbf{b}_i^{(1)}\right]d\mathbf{b}_i^{(2)}d\mathbf{b}_i^{(1)}d\boldsymbol{\beta} \tag{1.12}$$

Using $p(\mathbf{y}_{ij},\mathbf{b}_{ij}^{(2)}|\boldsymbol{\beta},\mathbf{b}_i^{(1)})$ from eq. 1.10 gives us:

$$\prod_{i=1}^{I}\int \exp\left[\sum_{j=1}^{nsub}\mathbf{y}_{ij}\cdot\boldsymbol{\eta}_{ij}-\sum_{j=1}^{nsub}\sum_{k=1}^{nobs}\log(1+\exp(\eta_{ijk}))-\sum_{j=1}^{nsub}\frac{1}{2}(\mathbf{b}_{ij}^{(2)})^T(\boldsymbol{\Sigma}^{(2)}(\boldsymbol{\phi}^{(2)}))^{(-1)}\mathbf{b}_{ij}^{(2)}\right]*$$
$$\exp\left[-\frac{1}{2}(\mathbf{b}_i^{(1)})^T(\boldsymbol{\Sigma}^{(1)}(\boldsymbol{\phi}^{(1)}))^{(-1)}\mathbf{b}_i^{(1)}\right]d\mathbf{b}_i^{(2)}d\mathbf{b}_i^{(1)}d\boldsymbol{\beta} \tag{1.13}$$

Finding ML or REML estimates through eqs. 1.9 and 1.12 requires large dimensional integrations with respect to the random effects. To perform these high dimensional integrations, we apply the spherical radial integration method of Clarkson et al [6] to each level of the hierarchy.

1.4 Whole SR and two SR

In this section, we introduce two methods of performing spherical radial integration: the whole SR method and the 2 SR method. To illustrate the differences between the whole SR and two SR method consider the following integral:

$$\begin{aligned}\int\int h(\mathbf{b}_i^{(2)},\mathbf{b}_i^{(1)})d\mathbf{b}_i^{(2)}d\mathbf{b}_i^{(1)} &= \int f(\mathbf{b}_i^{(1)})\left[\prod_{j=1}^{nsub}\int g_j(\mathbf{b}_{ij}^{(2)},\mathbf{b}_i^{(1)})d\mathbf{b}_{ij}^{(2)}\right]d\mathbf{b}_i^{(1)}\\ &= \int\int f(\mathbf{b}_i^{(1)})\left[\prod_{j=1}^{nsub} g_j(\mathbf{b}_{ij}^{(2)},\mathbf{b}_i^{(1)})\right]d\mathbf{b}_i^{(2)}d\mathbf{b}_i^{(1)}\end{aligned}$$

where $\mathbf{b}_i^{(2)} = (\mathbf{b}_{i,j=1}^{(2),T}, \ldots, \mathbf{b}_{i,j=nsub}^{(2),T})^T$. Notice that these integrals are of the same form of eqs. 1.9 and 1.12, but the number of groups has been set to one. In the whole SR method, we perform spherical radial integration one time to the whole integrand, $h(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)})$, whereas in the 2SR method we apply spherical radial integration to each of the inner integrals, $\int g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_{ij}^{(2)}$, with respect to $\mathbf{b}_{ij}^{(2)}$ and to the outer integrand, $f(\mathbf{b}_i^{(1)}) \left[\prod_{j=1}^{nsub} \int g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_{ij}^{(2)}\right]$, with respect to $\mathbf{b}_i^{(1)}$. However, one of the main assumptions of the spherical radial method is that the integrand must be approximately spherically symmetric, so we need to prove two lemmas: (1) if we transform $h(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)})$ by eq. 1.2 then $\int h(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_i^{(2)}$ is also spherically symmetric, and (2) we need to show that if $h(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)})$ is spherically symmetric then $g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)})$ is spherically symmetric with respect to $\mathbf{b}_{ij}^{(2)}$. Lastly, we will show that transforming h into a spherically symmetric function by eq. 1.2 preserves the independence structure of the data.

Lemma 1.1 *If $f(\mathbf{x}, \mathbf{y})$ is spherically symmetric than so is $g(\mathbf{y}) = \int f(\mathbf{x}, \mathbf{y}) d\mathbf{x}$.*

Proof. For this assertion, we consider an integral of the form:

$$\int \int f(\mathbf{x}, \mathbf{y}) d\mathbf{x} d\mathbf{y} \tag{1.14}$$

Let R be a rotation matrix (orthogonal and det=1) then we have:

$$g(R\mathbf{y}) = \int f(\mathbf{x}, R\mathbf{y}) d\mathbf{x} = \int f(\begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}) d\mathbf{x} \tag{1.15}$$

Since $\begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix}$ is orthogonal and has determinant one, it is a rotation matrix, so:

$$\int f(\begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0} & \mathbf{R} \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix}) d\mathbf{x} = \int f(\mathbf{x}, \mathbf{y}) d\mathbf{x} = g(\mathbf{y}) \tag{1.16}$$

which proves that $g(\mathbf{y})$ is spherically symmetric. ◇

Lemma 1.2 *If h is spherically symmetric, then $g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)})$ is spherically symmetric with respect to $\mathbf{b}_{ij}^{(2)}$.*

Proof. Let R be any rotation matrix. Consider an identity matrix blocked into diagonal blocks with sizes given by $(\mathbf{b}_{i,j=1}^{(2),T}, \ldots, \mathbf{b}_{i,j=nsub}^{(2),T}, \mathbf{b}_i^{(1),T})^T$. Then, let matrix, $\mathbf{G}$, be the matrix where the q^{th} diagonal block is replaced by R. Since G is orthogonal and has determinant one, it is a rotation matrix, so $h(\mathbf{G}\mathbf{b}_i) = h(\mathbf{b}_i)$ because h is spherically symmetric and $\mathbf{b}_i$ is defined as $(\mathbf{b}_i^{(2),T}, \mathbf{b}_i^{(1),T})^T$. Substituting the form of h into $h(\mathbf{G}\mathbf{b}_i) = h(\mathbf{b}_i)$ yields:

$$f(\mathbf{b}_i^{(1)}) \left[\prod_{j=1, j\neq q}^{nsub} g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) \right] g_q(R\mathbf{b}_{iq}^{(2)}, \mathbf{b}_i^{(1)}) = f(\mathbf{b}_i^{(1)}) \left[\prod_{j=1, j\neq q}^{nsub} g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) \right] g_q(\mathbf{b}_{iq}^{(2)}, \mathbf{b}_i^{(1)}) \tag{1.17}$$

Canceling, common terms gives us:

$$g_q(R\mathbf{b}_{iq}^{(2)}, \mathbf{b}_i^{(1)}) = g_q(\mathbf{b}_{iq}^{(2)}, \mathbf{b}_i^{(1)}) \tag{1.18}$$

which shows that g_q is spherically symmetric with respect to $\mathbf{b}_{iq}^{(2)}$. ◇

Lemma 1.3 *Transforming h into a spherically symmetric function by eq. 1.2 preserves the independence structure of the data.*

Proof. Consider the following integral:

$$\begin{aligned} \prod_{i=1}^{I} \int \int h(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_i^{(2)} d\mathbf{b}_i^{(1)} &= \prod_{i=1}^{I} \int f(\mathbf{b}_i^{(1)}) \left[\prod_{j=1}^{nsub} \int g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_{ij}^{(2)} \right] d\mathbf{b}_i^{(1)} \\ &= \prod_{i=1}^{I} \int \int f(\mathbf{b}_i^{(1)}) \prod_{j=1}^{nsub} g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_{ij}^{(2)} d\mathbf{b}_i^{(1)} \end{aligned}$$

Next, we transform $\mathbf{b}_i = (\mathbf{b}_i^{(2),T}, \mathbf{b}_i^{(1),T})^T$ into $\mathbf{x}_i = (\mathbf{x}_i^{(2),T}, \mathbf{x}_i^{(1),T})^T$ for each group i by $\mathbf{x}_i = \mathbf{B}_i(\mathbf{b}_i - \hat{\mathbf{b}}_i)$ where $\mathbf{B}_i$ is given by the Cholesky decomposition of the

Hessian: $\mathbf{H}_i = \mathbf{B}_i^T \mathbf{B}_i$ and $\mathbf{x}_i^{(2)} = (\mathbf{x}_{i,j=1}^{(2),T}, \ldots, \mathbf{x}_{i,j=nsub}^{(2),T})^T$. Under this transformation, we have:

$$\int \int h^*(\mathbf{x}_i) d\mathbf{x}_i = \int \int |\mathbf{B}_i^{-1}| h(\hat{\mathbf{b}_i} + \mathbf{B}_i^{-1}\mathbf{x}_i) d\mathbf{x}_i \tag{1.19}$$

Pinheiro and Chao [37] showed that each group's Hessian has the following block structure:

$$\mathbf{H_i} = \begin{pmatrix} () & & & & () \\ & () & & & () \\ & & \ddots & & \vdots \\ & & & () & () \\ () & () & \ldots & () & () \end{pmatrix} \tag{1.20}$$

It can be shown that $\mathbf{B}_i$ has the structure:

$$\mathbf{B_i} = \begin{pmatrix} () & & & & () \\ & () & & & () \\ & & \ddots & & \vdots \\ & & & () & () \\ & & & & () \end{pmatrix} \tag{1.21}$$

and therefore its inverse has the same block structure by the block matrix inverse formula. Let us define $\mathbf{b}_{i,new}^T$ as $(\mathbf{b}_{i,j=1,new}^{(2),T}, \ldots, \mathbf{b}_{i,j=nsub,new}^{(2),T}, \mathbf{b}_{i,new}^{(1),T})^T = \mathbf{b}_{i,new} = \hat{\mathbf{b}_i} + \mathbf{B}_i^{(-1)}\mathbf{x}_i$. Using the block structure of $\mathbf{B}_i^{-1}$ we see that $\mathbf{b}_{i,j,new}^{(2)}$ only depends on $\mathbf{x}_{ij}^{(2)}$ and $\mathbf{x}_i^{(1)}$. Furthermore, $\mathbf{b}_{i,new}^{(1)}$ only depends on $\mathbf{x}_i^{(1)}$, so we have shown that the independence structure is preserved after transforming h into an approximately spherically symmetric function. $\diamond$

Lastly, we compare the 2SR method with the whole SR method in terms of time complexity. Let us consider the following integral:

$$\prod_{i=1}^{I} \int f(\mathbf{b}_i^{(1)}) \left[\prod_{j=1}^{nsub} \int g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_{ij}^{(2)} \right] d\mathbf{b}_i^{(1)}$$

Counting the number of function evaluations for the whole SR method gives us $(nsub * q_2 + q_1 + 1)N_{quad} * I$ evaluations of f and $(nsub * q_2 + q_1 + 1)N_{quad} * nsub * I$ evaluations of g where N_{quad} is the number of radial quadrature points, q_2 is the number of random effects per subject, and q_1 is the number of level one random effects. In contrast, the 2 SR method has $N_{quad,outer}(q_1 + 1) * I$ evaluations of f and $(q_2 + 1)N_{quad,inner} * nsub * N_{quad,outer} * (q_1 + 1) * I$ evaluations of g where $N_{quad,outer}$ is the number of radial quadrature points for the outer integral and $N_{quad,inner}$ is the number of radial quadrature points for the inner integral. From this calculation, we observe that the whole SR is quadratic in the number of subjects, whereas 2 SR is only linear in the number of subjects. Furthermore, the 2SR method is linear in the number of random effects per subject, the number of level one random effects, and the number of groups. Later in section 1.6, we compare the estimation performance of our 2 SR method with the PQL approximation and with Gauss Hermite Quadrature.

1.5 Connections between two-level GLMM and GSMM

Let us consider a generalized semiparametric mixture model (GSMM) of the following form:

$$logit(\mu_{ij}) = \mathbf{x}_{ij}^T \boldsymbol{\beta} + \mathbf{a}_{ij}^T \mathbf{b}_i + \Theta(z_{ij}) + \Theta_i(z_{ij}) \tag{1.22}$$

Our response variable, $y_{i,j}$, for observation $j = 1 \ldots nsub_i$ of subject $i = 1 \ldots nsub$ is assumed to be binary. $\mathbf{x}_{i,j}$ in the above model represents a set of fixed effect covariates whereas $\mathbf{a}_{i,j}$ represents a set of random effect covariates. The vector, $\mathbf{b}_i$, are subject random effects with density: $\mathbf{b}_i \sim N(0, \Sigma_b)$ where the covariance

matrix, $\mathbf{\Sigma}$, equals $\begin{pmatrix} \phi_0 & 0 \\ 0 & \phi_1 \end{pmatrix}$. $\Theta(.)$ is a smooth function of the covariates, z_{ij}, and $\Theta_i(.)$ are subject specific smooth functions. Lastly, we assume that the subjects are independent, that $z_{i,j}$ is not included in $\mathbf{x}_{i,j}$ or $\mathbf{a}_{i,j}$, and that $\mathbf{b}_i$ and $\Theta_i(.)$ are independent.

Next, we approximate the smooth functions, $\Theta(.)$, and $\Theta_i(.)$ by a spline approximation. There are a variety of splines to choose from such as natural cubic splines, thin plate splines, or B-splines. Here, we adopt the truncated power series basis for our spline. The spline approximation of $\Theta(.)$ then takes the form:

$$\Theta(z_{ij}) = z_{ij}d + \mathbf{q}^T(z_{ij})\mathbf{c} \tag{1.23}$$

where our spline basis, $\mathbf{q}(z_{ij})$, is $(z_{i,j} - \kappa_k)_+, k = 1 \dots K$. κ_k are the knot locations and K is the number of knots. Our spline approximation of the subject smooth curves, $\Theta_i(.)$ is:

$$\Theta_i(z_{ij}) = z_{ij}e_i + \mathbf{q}^T(z_{ij})\mathbf{f}_i \tag{1.24}$$

where e_i has the prior, $N(0, \sigma_e^2)$, and $\mathbf{f}_i$ has the prior $N(0, \sigma_f^2\mathbf{I})$. Plugging these spline approximations into eq. 1.22 yields:

$$\eta_{ij} = logit(\mu_{ij}) = \mathbf{x}_{ij}^T\boldsymbol{\beta} + \mathbf{a}_{ij}^T\mathbf{b}_i + z_{ij}d + \mathbf{q}^T(z_{ij})\mathbf{c} + z_{ij}e_i + \mathbf{q}^T(z_{ij})\mathbf{f}_i \tag{1.25}$$

Lastly, we stack observations to yield:

$$\boldsymbol{\eta_i} = \begin{pmatrix} \mathbf{X}_i & \mathbf{z}_i \end{pmatrix} \begin{pmatrix} \boldsymbol{\beta} \\ d \end{pmatrix} + \mathbf{Q}_i\mathbf{c} + \begin{pmatrix} \mathbf{A}_i & \mathbf{z}_i & \mathbf{Q}_i \end{pmatrix} \begin{pmatrix} \mathbf{b}_i \\ e_i \\ \mathbf{f}_i \end{pmatrix} \tag{1.26}$$

where matrix $\mathbf{X}_i$ is the row stacked $\mathbf{x}_{ij}^T$'s, $\mathbf{z}_i = (z_{i,1}, \dots, z_{i,nsub_i})^T$, $\mathbf{Q}_i$ is the row-stacked $\mathbf{q}^T$'s, and $\mathbf{A}_i$ is the row stacked $\mathbf{a}_{ij}^T$'s.

For model fitting, we adopt the double penalized spline estimation approach of Lin and Zhang [28] to estimate $(\boldsymbol{\beta}, d, \mathbf{c})$. Estimates of $(\boldsymbol{\beta}, d, \mathbf{c})$ can be obtained by minimizing:

$$\sum_{i=1}^{nsub} p(\mathbf{y}_i, \mathbf{b}_i, e_i, \mathbf{f}_i) + \frac{1}{2\lambda}\mathbf{c}^T\mathbf{c} \tag{1.27}$$

where λ is a smoothing parameter controlling goodness of fit versus model complexity. $\mathbf{y}_i$ is defined as $(y_{i,1}, \ldots, y_{i,nsub_i})^T$, and the negative log joint distribution, $p(\mathbf{y}_i, \mathbf{b}_i, e_i, \mathbf{f}_i)$, equals:

$$-\mathbf{y}_i \cdot \boldsymbol{\eta}_i + \sum_j \log(1 + \exp(\eta_{i,j})) + \frac{1}{2}\mathbf{b}_i^T \begin{pmatrix} \phi_0^{-1} & 0 \\ 0 & \phi_1^{-1} \end{pmatrix} \mathbf{b}_i + \frac{e_i^2}{2\sigma_e^2} + \frac{\mathbf{f}_i^T\mathbf{f}_i}{2\sigma_f^2} \tag{1.28}$$

To estimate the smoothing parameter, λ, generalized cross validation (GCV), generalized maximum likelihood (GML), and unbiased risk (UBR) have been proposed. Here, we adopt GML due to its stability properties [8]. Extending Wabha's GML [44] approach to GSMM, we define the GML estimate of $(\lambda, \phi_0, \phi_1, \sigma_e^2, \sigma_f^2)$ as:

$$f(\mathbf{y}|\lambda, \phi_0, \phi_1, \sigma_e^2, \sigma_f^2) = \int \prod_{i=1}^{nsub} \left[\int \exp(-p(\mathbf{y}_i, \mathbf{b}_i, e_i, \mathbf{f}_i) d\mathbf{b}_i de_i d\mathbf{f}_i \right] \exp(-p(\mathbf{c})) d\mathbf{c} d\boldsymbol{\beta} dd \tag{1.29}$$

where $p(\mathbf{c})$ is the negative log density of a $N(0, \lambda\mathbf{I})$ and $\mathbf{y}$ is $(\mathbf{y}_1^T, \ldots, \mathbf{y}_{nsub}^T)^T$. Recognizing the integrand as the joint likelihood of $(\mathbf{y}, \mathbf{c})$, we can consider eq. 1.26 as a two-level GLMM with prior $\mathbf{c} \sim N(0, \lambda\mathbf{I})$. In this two-level GLMM model, the REML estimate of $(\lambda, \phi_0, \phi_1, \sigma_e^2, \sigma_f^2)$ is equivalent to the GML estimate defined above. Furthermore, we can define the ML estimates of $(\boldsymbol{\beta}, d, \lambda, \phi_0, \phi_1, \sigma_e^2, \sigma_f^2)$ as the maximizer of:

$$f(\mathbf{y}|\boldsymbol{\beta}, d, \lambda, \phi_0, \phi_1, \sigma_e^2, \sigma_f^2) = \int \prod_{i=1}^{nsub} \left[\int \exp(-p(\mathbf{y}_i, \mathbf{b}_i, e_i, \mathbf{f}_i) d\mathbf{b}_i de_i d\mathbf{f}_i \right] \exp(-p(\mathbf{c})) d\mathbf{c} \tag{1.30}$$

Notice that the ML and REML estimates of our GSMM model given by eqns. 1.29 and 1.30 are just a special case of the ML and REML estimates of a general

two-level GLMM when the number of groups is set to one. The general form of ML and REML estimates are given by eqns. 1.9 and 1.12. Furthermore, the negative log joint distribution, $p(\mathbf{y}_i, \mathbf{b}_i, e_i, \mathbf{f}_i)$, eq. 1.28, is a special case of eqn. 1.10 for the general two level model when the number of groups is one.

1.6 Simulation study for GLMMs and GSMMs

In this section, we performed a simulation study to compare the estimation performance of the spherical radial approximation versus the PQL approximation and Gauss Hermite Quadrature (GHQ). In our implementation of Gauss Hermite Quadrature for multilevel models, we applied a spherical transformation to the integrand to make the integrand approximately spherically symmetric, and then we applied Gauss Hermite product quadrature to each level of the hierarchy. We considered two applications of our proposed approach: the analysis of simulated multilevel GLMM models and the analysis of simulated GSMM models. We first considered the GLMM case and generated binary data from the following models:

$$logit(\mu_{ij}) = -3.0 + 2x_{2i} - x_{3ij} + b_{1i}x_{4i} + b_{2i}x_{5ij} + b_3x_{6i} + b_4x_{7ij} \tag{1.31}$$

$$logit(\mu_{ij}) = -2.0 + 2x_{2i} - x_{3ij} + b_{1i}x_{4i} + b_{2i}x_{5ij} + b_3x_{6i} + b_4x_{7ij} \tag{1.32}$$

$$logit(\mu_{ij}) = -2.5 + 2x_{2i} - x_{3ij} + b_{1i}x_{4i} + b_{2i}x_{5ij} + b_3x_{6i} + b_4x_{7ij} \tag{1.33}$$

Let the index i represent the subject index and let j be the jth observation of subject i. In our simulations, we set the sample designs to be 100 x 7, 100 x 5, and 100 x 7 respectively. b_{1i} and b_{2i} in our above models represent subject level random effects and are generated from a normal density with mean 0 and a true common variance of 1. b_3 and b_4 are population level random effects which are generated from a normal density with mean 0 and a true common variance of 1. Here, all of

the x's are generated from a uniform (-0.5,0.5).

For model fitting, we fitted models 1.31, 1.32, and 1.33 with a two-level GLMM model using a common variance of VC_{sub} for the subject level random effects and a common variance of VC_{pop} for the population level random effects. The following tables are based on 100 repeated simulations. We report the mean and the standard error across 100 simulations for the fixed effects β_1 thru β_3 as well as the mean squared error (MSE) of each of the fixed effects. The rows of the table compare the estimation performance of the PQL approximation with the spherical radial maximum likelihood approach and with the GHQ approach.

We also compare the estimation performance for the variance components. We report the mean and standard error across 100 simulations for the variance components, VC_{sub} and VC_{pop}. The columns VC sub MSE, VC pop MSE, and overall MSE represent the mean squared error of the subject level variance component, the mean squared error of the population level variance component, and the mean squared error of the overall fit respectively.

Table 1. Fixed effects of GLMM model 1.31

Method	β_1	β_1 MSE	β_2	β_2 MSE	β_3	β_3 MSE
Truth	-3		2		-1	
ML	-3.076 (0.236)	0.062	2.110 (0.670)	0.461	-0.961 (0.628)	0.396
PQL	-3.320 (0.418)	0.278	2.195 (0.850)	0.761	-1.045 (0.752)	0.567
GHQ	-3.092 (0.242)	0.0671	2.121 (0.682)	0.479	-0.981 (0.621)	0.385

Table 2. Variance components of GLMM model 1.31

Method	VC sub	MSE	VC pop	MSE	overall MSE
Truth	1		1		
ML	0.972 (1.295)	1.677	1.631 (1.730)	3.390	0.343
PQL	1.449 (1.745)	3.247	11.667 (9.171)	197.885	1.234
GHQ	0.936 (1.328)	1.768	1.897 (1.745)	3.849	0.353

Table 3. Fixed effects of GLMM model 1.32

Method	β_1	β_1 MSE	β_2	β_2 MSE	β_3	β_3 MSE
Truth	-2		2		- 1	
ML	-2.026 (0.164)	0.028	2.094 (0.537)	0.297	-0.908 (0.470)	0.229
PQL	-2.030 (0.180)	0.033	2.096 (0.533)	0.294	-0.926 (0.489)	0.244
GHQ	-2.026 (0.165)	0.028	2.091 (0.542)	0.302	-0.908 (0.470)	0.229

Table 4. Variance components of GLMM model 1.32

Method	VC sub	MSE	VC pop	MSE	overall MSE
Truth	1		1		
ML	1.065 (1.492)	2.229	0.952 (1.300)	1.693	0.270
PQL	1.157 (1.599)	2.583	3.552 (3.911)	21.806	0.403
GHQ	1.087 (1.498)	2.251	0.965 (1.296)	1.680	0.271

Table 5. Fixed effects of GLMM model 1.33

Method	β_1	β_1 MSE	β_2	β_2 MSE	β_3	β_3 MSE
Truth	-2.5		2		- 1	
ML	-2.551 (0.183)	0.036	2.084 (0.557)	0.317	-0.954 (0.402)	0.164
PQL	-2.595 (0.227)	0.061	2.103 (0.568)	0.333	-0.966 (0.389)	0.153

Table 6. Variance components of GLMM model 1.33

Method	VC sub	MSE	VC pop	MSE	overall MSE
Truth	1		1		
ML	1.168 (1.213)	1.500	1.230 (1.456)	2.173	0.264
PQL	1.309 (1.310)	1.812	4.837 (4.401)	34.097	0.493

The tables show that the ML method has favorable performance compared to the PQL approximation. Notice that the population variance component estimate is dramatically better when using the ML method. In addition, the ML method shows a smaller MSE for the fixed effect estimates and for the subject level variance component estimate. As for Gauss Hermite Quadrature, the spherical radial approximation has similar to better performance.

The second application of our approach is the analysis of generalized semiparameteric mixed models (GSMMs). To this end, we generated binary data from

the following GSMM model:

$$logit(\mu_{ij}) = -0.5 + b_{0i} + b_{1i}t_{1ij} + 8t_{2ij}^2 + b_{2i}t_{2ij} + 3b_{3i}t_{2ij}^2$$
$$b_{0i} \sim N(1.2, 1), b_{1i} \sim N(12, 1), b_{2i} \sim N(-5, 2), b_{3i} \sim N(5, 2) \tag{1.34}$$

Here, t_{1ij} and t_{2ij} were generated from a Uniform (-0.5,0.5). We fitted the GSMM model by eq. 1.26 using 10 knots and a spline basis of q(x) $= (x-\kappa_j)_+, j = 1 \ldots K$. Knots locations were given by the quantiles of t_{2ij}. In eq. 1.26, we set matrix $\mathbf{X}_i$ to equal matrix $\mathbf{A}_i$ with both matrices having two columns: $\mathbf{1}$ and $\mathbf{t}_1$ where $\mathbf{t}_1$ is the vector of t_{1ij}.

The following tables are based on 100 repeated simulations for one sample design: 33x17. In the following, Coef t_1 represents the mean(standard deviation) of the fixed coefficient of t_1. MSE is the mean squared error in the estimate of the fixed coefficient of t_1. VC t_1 represents the variance component's mean and standard deviation. We also report the variance component's mean squared error under the label, MSE. Furthermore, we report MSE_P, MSE_I, and overall MSE which are the mean squared error of the estimated population curve of t_2, the mean squared error for the estimated individual curves, and the mean squared error of the overall fit respectively.

Table 7. GSMM results

Method	Coef t_1	MSE	VC t_1	MSE
ML	12.239 (1.149)	1.378	2.464 (3.494)	14.355
PQL	12.651 (2.371)	6.045	11.217 (10.891)	223.015

Table 8. GSMM results continued...

Method	$MSE_P t_2$	$MSE_I t_2$	overall MSE
ML	1.480	3.808	4.009
PQL	1.463	4.199	5.127

The above table shows that the ML method has a smaller MSE compared to the PQL method for the fixed coefficient of t_1 and for the variance component, VC t_1. Additionally, we see that ML has a smaller $MSE_I t_2$ and a smaller overall MSE when compared to the PQL method.

1.7 HSR package

In this section, we describe the Hierarchical Spherical Radial package (HSR) that we developed to perform the spherical radial calculations. Spherical radial integration is applied to each level of the hierarchy, namely it is applied to each of the inner integrals as well as the outer integral of:

$$\prod_{i=1}^{I} \int f(\mathbf{b}_i^{(1)}) \left[\prod_{j=1}^{nsub} \int g_j(\mathbf{b}_{ij}^{(2)}, \mathbf{b}_i^{(1)}) d\mathbf{b}_{ij}^{(2)} \right] d\mathbf{b}_i^{(1)} \tag{1.35}$$

Let $h(\mathbf{x})$ be a n-dimensional function to be integrated for one of the two levels. The package first transforms $h(\mathbf{x})$ into an approximately spherically symmetric function by eqn. 1.2, and then integrates $h(\mathbf{x})$ by eqn. 1.3. The radial integral is calculated by a Gauss-Kronrod rule, whereas the polar integral is calculated by eqn. 1.4.

The HSR package can be used to analyze two level GLMMs for a binary response as well as GSMMs for a binary response. In the case of the GLMMs, the first step is to load the package:

library(HSR)

Next, we set the number of subjects and the number of observations per subject.

nsub ← 10

nobs.persub ← 20

The dataset is then read and put into a dataframe.

data ← read.table("c:/profliu/AIDS/multilevel10x20RE2x2.txt")

index ← 1:200

x2 ← data[index, 4]

x3 ← data[index, 5]

x4 ← data[index, 6]

x5 ← data[index, 7]

x6 ← data[index, 8]

x7 ← data[index, 9]

y ← data[index, ncol(data)]

mydata ← data.frame(x2, x3, x4, x5, x6, x7, y)

The user can choose the number of radial Gauss-Kronrod points for the inner and outer integrals of eqn. 1.35:

GKlevelone ← 3

GKleveltwo ← 3

3, 5, 7, 15, 31, and 61 points are available options. The user can also control the number of radial rotations, N_r, and the initial guesses for the fixed effects, β, the variance components, σ_0^2, and random effects, bhat0:

Nr ← 1

beta0 ← rep(0, 3)

sigma20 ← rep(0.1, 2)

bhat0 ← rep(0, 2 * 10 + 2)

Note that the random effects are stacked by $(\mathbf{b}_{i,j=1}^{(2),T}, \ldots, \mathbf{b}_{i,j=nsub}^{(2),T}, \mathbf{b}_i^{(1),T})^T$. Unfortunately, the code only handles the $i = 1$ at present. For the fixed effect covariate

matrix and the random effect covariate matrix, the user sets the variables X, Zlevelone, and Zleveltwo.

X ← cbind(1, x2, x3)

Zlevelone ← cbind(x6, x7)

Zleveltwo ← cbind(x4, x5)

Other user parameters are Urlevelone and Urleveltwo. Integration over r requires a finite range of integration, which is controlled by these two parameters.

Urlevelone ← 4 * sqrt(ncol(Zlevelone) - 1)

Urleveltwo ← 4 * sqrt(ncol(Zleveltwo) - 1)

Output is controlled by the variable, outfile:

outfile ← "c:/profliu/AIDS/testglmm.R"

If the user wishes to set a time limit on the amount of time to perform ML optimization, the cuttime parameter is set:

cuttime ← 3600

Lastly, one sets up the group structure of the covariance matrix and calls the hsr function:

grpleveltwo ← rep(1, 2)

grplevelone ← rep(1, 2)

hsrxy ← HSR(X, y, Zlevelone, Zleveltwo, GKlevelone, GKleveltwo, nsub, nobs.persub, beta0, sigma20, cuttime, outfile, grplevelone, grpleveltwo, Nr, bhat0, Urlevelone, Urleveltwo)

The covariance structure of the random effects, at present, assumes a diagonal covariance structure. In this package, the grouping index must start at index 1

and by followed by index 2, 3, 4, an so on. Any variance parameter with the same grouping index will have a common variance. An alternative to the matrix interface to hsr is the formula interface:

hsrtilda ← HSR(y ∼ x2 + x3, data = mydata, Zlevelone, Zleveltwo, GKlevelone, GKleveltwo, nsub, nobs.persub, beta0, sigma20, cuttime, outfile, grplevelone, grpleveltwo, Nr, bhat0, Urlevelone, Urleveltwo)

For reference, the hsr usage summary is below:

Call:
HSR.default(x = X, y = y, Zmatlevelone = Zlevelone, Zmatleveltwo = Zleveltwo, GKlevelone = GKlevelone, GKleveltwo = GKleveltwo, nsub = nsub, nobs.persub = nobs.persub, beta0 = beta0, sigma20 = sigma20, cuttime = cuttime, outfile = outfile, grplevelone = grplevelone, grpleveltwo = grpleveltwo, Nr = Nr, bhatav = bhat0, Urlevelone = Urlevelone, Urleveltwo = Urleveltwo)

Call:
HSR.formula(formula = y ∼ x2 + x3, data = mydata, Zlevelone, Zleveltwo, GKlevelone, GKleveltwo, nsub, nobs.persub, beta0, sigma20, cuttime, outfile, grplevelone, grpleveltwo, Nr, bhat0, Urlevelone, Urleveltwo)

Besides two-level GLMMs, the package can be used to analyze GSMM models. In our example, we approximate the non-parametric functions by a truncated power series basis:

num.knots ← 10
knots2 ← quantile(unique(x2),
seq(0, 1, length = (num.knots + 2))[-c(1,(num.knots + 2))])

Dmat ← pmax(outer(knots2, knots2,"-"), 0)

Qmat ← pmax(outer(x2, knots2,"-"), 0)

As we have shown in section 1.5, our GSMM can be formulated as a two-level GLMM (see eqn. 1.26). With this representation, the fixed effect covariate matrix and the random effect covariate matrices are:

Zlevelone ← Qmat

Zleveltwo ← cbind(1, x1, x2, Qmat)

X ← cbind(1, x1, x2)

We also need to alter the grouping structure:

grpleveltwo ← c(1:4, rep(4, num.knots - 1))

grplevelone ← rep(1, num.knots)

Consequently, the level two covariance matrix is diagonal with entries: $(\sigma_1^{2,(2)}, \sigma_2^{2,(2)}, \sigma_3^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)}, \sigma_4^{2,(2)})$ and the level one covariance matrix is diagonal with entries $(\sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)}, \sigma_1^{2,(1)})$. For completeness, I list the entire sample GSMM code below:

library(HSR)

 # GSMM example code

 # number of subjects, number of observations per subject, and # of spline knots

nsub ← 33

nobs.persub ← 17

num.knots ← 10

```
# read the dataset
data ← read.table("c:/profliu/AIDS/coreid1/M3data33x17.txt")
index ← 1:(33 * 17)
x1 ← data[index, 1]
x2 ← data[index, 2]
y ← data[index, ncol(data)]
mydata ← data.frame(x1, x2, y)

# quadrature points
GKlevelone ← 3
GKleveltwo ← 3

# number of radial rotations
Nr ← 1

# form the truncated power series
knots2 ← quantile(unique(x2),
seq(0, 1, length = (num.knots + 2))[-c(1,(num.knots + 2))])
Dmat ← pmax(outer(knots2,knots2,"-"), 0)
Qmat ← pmax(outer(x2,knots2,"-"), 0)

# setup matrices
Zlevelone ← Qmat
Zleveltwo ← cbind(1, x1, x2, Qmat)
```

```
# initial guess for beta, sigma, and bhatl sigma on log scale
beta0 ← rep(0, 3)
sigma20 ← rep(0.1, 5)
bhat0 ← rep(0, 13 * 33 + 10)

# integration cutoffs
Urlevelone ← 4 * sqrt(ncol(Zlevelone) - 1)
Urleveltwo ← 4 * sqrt(ncol(Zleveltwo) - 1)

# max time for optimization

cuttime ← 3600

# output file
outfile ← "c:/profliu/AIDS/testgsmm.R"

# group structure of sigmalevelone, sigmaleveltwo
grpleveltwo ← c(1:4, rep(4, num.knots - 1))
grplevelone ← rep(1, num.knots)

X ← cbind(1, x1, x2)

# matrix interface
resXY ← HSR(X, y, Zlevelone, Zleveltwo, GKlevelone, GKleveltwo, nsub,
nobs.persub, beta0, sigma20, cuttime, outfile, grplevelone, grpleveltwo,
Nr, bhatav, Urlevelone, Urleveltwo)
```

 # formula interface

restilda ← HSR(y ∼ x1 + x2, data = mydata, Zlevelone, Zleveltwo, GKlevelone, GKleveltwo, nsub, nobs.persub, beta0, sigma20, cuttime, outfile, grplevelone, grpleveltwo, Nr, bhatav, Urlevelone, Urleveltwo)

1.8 Future directions

One question of interest, for future research, is the impact of the misspecification of the random effects in GLMMs. Litière et al [24] studied GLMMs under misspecification by performing numerous simulation studies under a variety of variances and a variety of densities for data generation. He showed that maximum likelihood estimators are inconsistent when the random effects are misspecified [24]. He also demonstrated that the fixed effects have a small bias when the random effect variances are small, but the estimates of the variance components are always severely biased. The magnitude of the bias was shown to increase with large variances, and the model fits worsen as the number of random effects increase. Furthermore, the misspecification can greatly impact the power of the analysis. Some methods we could employ to study misspecification include the heterogeneity method [24], sensitivity analysis [24], skew extensions to the normal and the t-distribution [23], or semi-parametric random effect distributions [3].

A second area of interest is hypothesis testing, where we hope to investigate hypothesis testing by using the likelihood ratio test and by using parametric bootstrapping to generate the reference distributions. Lastly, we hope to apply our methodology to other types of data sets in medical and health services research. For example, our methods could be used to analyze survival models with frailty as well as two part random effect models for semi-continuous data.

CHAPTER 2

GENETIC PATHWAY MODELING

2.1 Introduction

The invention of the DNA microarray has led to an explosion in the availability of genomic data sets. We focus our analysis on gene sets/gene pathways rather than traditional individual gene analysis because differential expression of gene categories tends to be more reproducible across studies and easier to interpret biologically [30]. Previous approaches to gene set analysis include gene set analysis (GSA) of Efron and Tibshirani [9], Gene Set Enrichment Analysis (GSEA) of Subramanian et al [42], the global test of Goeman et al [14], logistic kernel machines from Dawei Liu et al [29], least squares kernel machines of Dawei Liu et al [30], and the multivariate analysis method of Nettleton et al [35].

A linear model approach to modeling a genetic pathway effect has been considered by Goeman et al [14]. They take the approach of a generalized linear model where each gene enters the model linearly. The regression coefficients of the gene are assumed to come from a common distribution with mean 0 and variance τ^2. Testing the genetic pathway effect is equivalent to treating all of the regression coefficients being zero. Based on the distributional assumption of the regression coefficients, we just need to test $\tau^2 = 0$.

Our approach to modeling a genetic pathway effect differs from Goemen et al [14] since we do not assume a linear genetic pathway effect, instead we model the pathway effect non-parametrically. Some non-parametric approaches include cubic splines, thin-plate splines, B-splines, penalized splines, or kernel machines. Here we adopt the kernel machine approach since they are highly flexible and can model linear effects, quadratic effects, interaction between genes, and non-linear effects. A linear model approach is unable to model all of these types of effects.

Another approach to perform gene set analysis is the multivariate analysis method of Nettleton et al [35]. In their approach gene expression measurements are represented by Y_{ijk} which is the gene expression measurement for treatment i, replication j, and gene k of a pathway of interest. He assumes that $\mathbf{Y}_{i,j} = (Y_{i,j,1} \ldots Y_{i,j,G})$ has a multivariate distribution F_i where G is the number of genes in the pathway of interest. He then tests for pathway differential expression by testing $H_0 : F_1 = \ldots = F_T$ where T is the number of treatments using the multi-response permutation procedure (MRPP) test of Mielke and Berry [33]. One main shortcoming of this approach is that the model does not control for clinical covariates, so we would like to create a model which tests for differential pathway expression controlling for clinical covariates.

We propose a logistic kernel machine to model a genetic pathway effect with a binary response. The logistic kernel machine approach allows us to model the genetic pathway effect taking into account interactions between genes as well as controlling for clinical covariates. Furthermore, we established a connection between our logistic kernel machine model with generalized linear mixed models (GLMM) allowing one to use ideas from the GLMM literature.

Here, we adopt the spherical radial approach to perform the high dimensional integrations required for estimation of the fixed effects in the model and the testing

of the genetic pathway effect. Previous approaches to test the pathway effect include the score test of [29], the GSA method [9], the GSEA method [42], and the MRPP test [33]. The GSEA method begins by calculating a gene statistic for all of the genes in the data. Next, they compute a gene-set statistic by comparing the gene statistics of a given category with the gene statistics outside of the given category. A signed Kolmogorov-Smirnov statistic is used to calculate a gene set score for each of the gene sets. A permutation test is then applied to calculate the significance of an observed gene set score. One problem with this approach is that the gene sets compete with one another for significance. This type of testing is called competitive testing, and we would like to develop a testing approach where gene sets do not compete with one another. This alternative approach is called self-contained testing [13]. For testing the genetic pathway effect, we propose a REML likelihood ratio test. This methodology has three main advantages over previous approaches: 1) our testing is self-contained rather than competitive, 2) our kernel machine approach can model complex pathway effects and gene-gene interactions, and 3) we test for the pathway effect adjusting for clinical covariates.

This chapter is organized as follows. In section 2.2, we discuss our logistic kernel machine model. Then, in section 2.3, we derive estimates of the fixed effects of the model and present our REML likelihood ratio test. In section 2.4, we perform a simulation study to compare our estimation approach with a Bayesian approach. The performance of our REML likelihood ratio test is compared with a score test in section 2.5. Lastly, in sections 2.6 and 2.7, we discuss an application of our approach to an Acute Lymphocytic Leukemia data set and discuss some future directions for our work.

2.2 The semiparametric model

Let Y_i be a binary response variable for subject i $(i = 1 \dots n)$ where μ_i is the probability of $Y_i = 1$ for subject i. Furthermore, let $\mathbf{x}_i$ be a vector of clinical covariates, β be a vector of fixed effects, $h(.)$ be a unknown smooth function representing a genetic pathway effect, and $\mathbf{z}_i$ be a q x 1 vector of the gene expression measurements within a pathway. We consider the following logistic semi-parametric model:

$$logit(\mu_i) = \mathbf{x}_i^T \boldsymbol{\beta} + h(\mathbf{z}_i) \tag{2.1}$$

Our genetic pathway function, $h(.)$, can be modeled parametrically or nonparametrically. Here, we model the genetic pathway effect non-parametrically by the use of kernel machines and assume that $h(.)$ lies in the function space, H_K, the function space generated by a positive definite kernel function, $K(.,.)$. We use the dual representation of $h(.)$ in H_K to write $h(\mathbf{z}_i)$ as $h(\mathbf{z}_i) = \sum_{j=1}^{n} c_j K(\mathbf{z}_i, \mathbf{z}_j; \rho)$.

Two common kernel functions are the polynomial kernel and the Gaussian kernel. The d^{th} polynomial kernel takes the form: $Ker(\mathbf{z}_1, \mathbf{z}_2) = (\mathbf{z}_1^T \mathbf{z}_2 + \rho)^d$, where ρ and d are kernel parameters. For $d = 1$, the polynomial kernel generates a linear function space, and for $d = 2$, we have a quadratic function space which models main effects, all of the two-way interactions, and quadratic effects. Another popular kernel function is the Gaussian kernel with the form: $Ker(\mathbf{z}_1, \mathbf{z}_2) = exp(-\frac{||\mathbf{z}_1 - \mathbf{z}_2||^2}{\rho})$ where $||\mathbf{z}_1 - \mathbf{z}_2||^2 = \sum_{j=1}^{q} (z_{1j} - z_{2j})^2$. Other options for kernel functions include B-splines and the sigmoid.

2.3 Estimation and testing in the semiparametric model

Modeling the non-parametric function with a kernel machine gives us the following logistic semi-parametric model:

$$\eta_i = logit(\mu_i) = \mathbf{x}_i^T\boldsymbol{\beta} + \sum_{j=1}^{n} c_j Ker(\mathbf{z}_i, \mathbf{z}_j; \rho, d) \tag{2.2}$$

Stacking subjects gives us: $\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} + \mathbf{F}\mathbf{c}$ where $\mathbf{F} = Ker(\mathbf{z}_i, \mathbf{z}_j; \rho, d)$. We then find estimates of $(\boldsymbol{\beta}, \mathbf{c})$ by minimizing the penalized negative log likelihood of $\mathbf{y}$:

$$\sum_{i=1}^{n} [-y_i\eta_i + \log(1 + \exp(\eta_i))] + \frac{1}{2\lambda}\mathbf{c}^T F\mathbf{c} \tag{2.3}$$

Extending Wahba's (1990) GML approach [44] to the kernel machine context allows us to form GML estimates of (λ, ρ) as the maximizer of:

$$\int \exp\left[-\sum_{i=1}^{n}(-y_i\eta_i + \log(1 + \exp(\eta_i)))\right] \frac{1}{(2\pi)^{\frac{n}{2}}} \frac{1}{|\lambda\mathbf{F}^-|^{\frac{1}{2}}} \exp(-\frac{1}{2\lambda}\mathbf{c}^T F\mathbf{c}) d\mathbf{c} d\boldsymbol{\beta} \tag{2.4}$$

Notice that the integrand of eq. 2.4 is the joint likelihood of $(\mathbf{y}, \mathbf{c})$ with $\mathbf{c} \sim N(0, \lambda\mathbf{F}^-)$. From this observation, we can then view our semiparametric model as a GLMM with $\mathbf{c}$ as random effect with prior $\mathbf{c} \sim N(0, \lambda\mathbf{F}^-)$. Then the above GML estimates of (λ, ρ) is also the REML estimates of (λ, ρ) in this GLMM. Furthermore, we define ML estimates of $(\boldsymbol{\beta}, \lambda, \rho)$ as the maximizer of:

$$\int \exp\left[-\sum_{i=1}^{n}(-y_i\eta_i + \log(1 + \exp(\eta_i)))\right] \frac{1}{(2\pi)^{\frac{n}{2}}} \frac{1}{|\lambda\mathbf{F}^-|^{\frac{1}{2}}} \exp(-\frac{1}{2\lambda}\mathbf{c}^T F\mathbf{c}) d\mathbf{c} \tag{2.5}$$

As for testing of the genetic pathway effect, we propose a REML based likelihood ratio test. Our null hypothesis for testing the genetic pathway effect is $H_0 : h = 0$ versus $H_A : h \neq 0$. Recognizing eq. 2.2 as a GLMM model with prior $\mathbf{c} \sim N(0, \lambda\mathbf{F}^-)$, we can recast our hypothesis test as: $H_0 : \lambda = 0$ vs its alternative. The REML likelihood ratio test then takes the form:

$$\frac{sup_\rho L(\lambda = 0, \rho, d|\mathbf{y})}{sup_{\lambda,\rho} L(\lambda, \rho, d|\mathbf{y})} \tag{2.6}$$

The likelihood under the null model is:

$$L(\lambda = 0, \rho, d|\mathbf{y}) = \int \exp(-p(\mathbf{y}, \boldsymbol{\beta}|\lambda = 0, \rho, d))d\boldsymbol{\beta} \tag{2.7}$$

where $p(\mathbf{y}, \boldsymbol{\beta}|\lambda = 0, \rho, d)$ is:

$$-\mathbf{y} \cdot \boldsymbol{\eta} + \sum_j \log(1 + \exp(\eta_j)) \tag{2.8}$$

The full model likelihood is given by:

$$L(\lambda, \rho, d|\mathbf{y}) = \int \int \exp(-p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{c}(\lambda), \rho, d)) \exp(-p(\mathbf{c}))d\boldsymbol{\beta}d\mathbf{c} \tag{2.9}$$

where $p(\mathbf{y}|\boldsymbol{\beta}, \mathbf{c}(\lambda), \rho, d)$ is given by:

$$p(\mathbf{y}|\mathbf{c}, \boldsymbol{\beta}, \lambda, \rho, d) = \sum_{j=1}^{n} [-y_j \eta_j + \log(1 + \exp(\eta_j))] \tag{2.10}$$

Significance of the REML likelihood ratio test statistics is then determined by the null distribution of test statistics generated from parametric bootstrapping. Unfortunately, finding ML or REML estimates through eqs. 2.4 and 2.5 or finding the REML likelihood ratio through eq. 2.6 requires large dimensional integrations with respect to the random effects. Here, we apply the spherical radial integration method of [6] to perform these high dimensional integrations.

2.4 Simulation study for parametric effects

In this section, we performed a simulation study to compare the estimation performance of ML based spherical radial and REML based spherical radial with a Bayesian approach. We simulated binary data from the following models with a 5-gene genetic pathway effect:

$$logit(\mu_i) = -5.0 + 0.10age_i + \mathbf{Z}(0.06, 0.14, 0.14, 0.06, 0.14)^T \tag{2.11}$$

$$
\begin{aligned}
logit(\mu_i) &= -5.0 + 0.10age_i + 0.07\mathbf{Z}_{i,1}^2 + 0.1\mathbf{Z}_{i,2} + \\
&\quad 0.035\mathbf{Z}_{i,3}^2 + 0.07\mathbf{Z}_{i,4} + 0.1\mathbf{Z}_{i,5} \qquad (2.12)
\end{aligned}
$$

$$
\begin{aligned}
logit(\mu_i) &= -5.0 + 0.10age_i + 0.07\mathbf{Z}_{i,1}^2 + 0.1\sin(\mathbf{Z}_{i,2}) + \\
&\quad 0.035\mathbf{Z}_{i,3}^2 + 0.07\mathbf{Z}_{i,4} + 0.1\mathbf{Z}_{i,5} \qquad (2.13)
\end{aligned}
$$

$$
\begin{aligned}
logit(\mu_i) &= -5.0 + 0.10age_i + 0.04\mathbf{Z}_{i,1}^2 + 0.1\sin(\mathbf{Z}_{i,2}) + \\
&\quad 0.05\mathbf{Z}_{i,3}^2 + 0.04\exp(0.5\mathbf{Z}_{i,4}) + 0.1\mathbf{Z}_{i,5} \qquad (2.14)
\end{aligned}
$$

$$
\begin{aligned}
logit(\mu_i) &= -5.0 + 0.10age_i + 0.18\mathbf{Z}_{i,1}^2\sin(\mathbf{Z}_{i,2}) + \\
&\quad 0.12\mathbf{Z}_{i,3}^2 + 0.12\mathbf{Z}_{i,4} + 0.12\mathbf{Z}_{i,5} \qquad (2.15)
\end{aligned}
$$

The above models represent genetic pathway effects that are linear, quadratic, trigonometric, exponential, and interaction effects respectively. The age covariate was generated from a discrete uniform with ages from 35 to 55. The true coefficient of age is set to be 0.1. 100 simulations were run with a sample size of 100. Gene expression measurements, $\mathbf{Z}_{i,j}$, for 5 genes, $j = 1 \ldots 5$, and for the first 50 samples, $i = 1 \ldots 50$, were generated from a multivariate normal, $N(\boldsymbol{\mu} = (0,0,0,0,0), \boldsymbol{\Sigma} = AR_1(\rho = 0.8))$. AR_1 represents an autoregressive-1 covariance structure. Samples 51-100 were generated from a multivariate normal with mean, $\boldsymbol{\mu} = (2.5, 2.5, 2.5, 0, 0)$, and covariance structure of an $AR_1(\rho = 0.8)$ process.

In the following tables we fitted the above models with a logistic kernel machine, eq. 2.2, using a Gaussian kernel with the kernel parameter set to 5. The tables compare the estimation performance of ML-based spherical radial and REML-based spherical radial with two implementations of Gibbs sampling: the Gibbs sampling

of Winbugs and the Gibbs sampling algorithm of Zeger et al [47]. The Gibbs sampler used 5000 iterations, a thinning parameter of 10, and 2500 iterations of burnin, whereas Winbugs used 10000 iterations, a thinning parameter of 5, and a burnin period of 5000 iterations. The non-parametric functions in the Gibbs sampler uses the method of Kim and Gu whereby the kernel machine uses a subset of the sample as knots [46]. This approach takes the form:

$$\eta_i = logit(\mu_i) = \mathbf{x}_i^T\boldsymbol{\beta} + \sum_{j=1}^{K} c_j Ker(\mathbf{z}_i, \boldsymbol{\kappa}_j; \rho, d) \tag{2.16}$$

Stacking subjects allows us to rewrite our semi-parametric model as:

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} + \mathbf{F}\mathbf{c} \tag{2.17}$$

where $\mathbf{F}_{a,b} = Ker(\mathbf{z}_a, \boldsymbol{\kappa}_b; \rho, d)$. We then recast $\boldsymbol{\eta}$ as:

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} + \mathbf{G}_1\mathbf{c}_{11} + \mathbf{G}_2\mathbf{c}_{12} \tag{2.18}$$

by first performing the spectral decomposition of $Ker(\boldsymbol{\kappa}_m, \boldsymbol{\kappa}_n; \rho, d) = \mathbf{K}(\rho, d) = \mathbf{U}\boldsymbol{\Lambda}\mathbf{U}^T$, where the eigenvalues and eigenvectors of $\mathbf{K}$ have been ordered so that $\boldsymbol{\Lambda} = diag(d_1, d_2, ...d_{NNZE}, 0, \ldots, 0)$ and NNZE is the number of non-zero eigenvalues of $\mathbf{K}$. Next, define $\mathbf{G}$ as $\mathbf{FU}$, $\mathbf{G}_1$ as the first NNZE columns of $\mathbf{G}$, $\mathbf{c}_1$ as $\mathbf{U}^T\mathbf{c}$, $\mathbf{c}_{11}$ as the first NNZE elements of $\mathbf{c}_1$, and $\mathbf{c}_{12}$ from $\mathbf{c}_1^T = (\mathbf{c}_{11}^T, \mathbf{c}_{12}^T)$. Our final equation for η is:

$$\boldsymbol{\eta} = \mathbf{X}\boldsymbol{\beta} + \mathbf{G}_1\Lambda_{new}^{-\frac{1}{2}}\mathbf{c}_{11,new} + \mathbf{G}_2\mathbf{c}_{12} \tag{2.19}$$

where $\mathbf{c}_{11,new}$ is $\boldsymbol{\Lambda}_{new}^{\frac{1}{2}}\mathbf{c}_{11}$ and $\boldsymbol{\Lambda}_{new}^{\frac{1}{2}} = diag(\sqrt{d_1}, \sqrt{d_2}, ..., \sqrt{d_{NNZE}})$.

For Winbugs, the fixed effect parameters of $\boldsymbol{\beta}$ and $\mathbf{c}_{12}$ have priors of independent normals with mean, $\mu = 1$, and variance of 10^5. $\mathbf{c}_{11,new}$ is modeled as a $N(0, \lambda\mathbf{I})$ with the inverse of λ modeled as a Gamma with shape parameter 10^{-5} and scale parameter 10^5. For Zeger's Gibbs sampler, the fixed effects $(\boldsymbol{\beta}, \mathbf{c}_{12})$ have a flat prior,

$\mathbf{c}_{11,new}$ has a prior of a $N(0, \lambda\mathbf{I})$, and λ has an inverse Gamma distribution with shape parameter = .001 and scale of .001. Estimation performance is compared by the SSR criterion where SSR = $\sum(\eta_i - \hat{\eta}_i)^2$.

The columns of the tables illustrate SSR, the computation time in hours on a single core of the quad core Q6600 processor, the estimate of the age effect, the bias in the age estimate, the standard error in the estimate of the age coefficient, and the number of simulations that didn't converge which we call "trap". The tables below show that maximum likelihood based on the spherical radial approximation has comparable to better goodness of fit by the SSR criterion when compared to the two implementations of Gibbs sampling. Additionally, the ML approach has a much lower computational cost than the Gibbs sampling approaches.

Table 9. 5-gene estimation: linear pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	33.099	58.101	0.099	-0.001	0.041	4
Winbugs	39.4971	26.958	0.0997	-0.0003	0.0419	1
REML	36.8653	0.269	0.0959	-0.0041	0.0404	0
ML	34.2584	1.066	0.0979	-0.0021	0.0406	1

Table 10. 5-gene estimation: quadratic pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	40.882	54.995	0.114	0.014	0.046	7
Winbugs	46.9579	26.998	0.1143	0.0143	0.0464	3
REML	41.1551	0.471	0.1093	0.0093	0.0439	1
ML	40.0527	0.893	0.112	0.0117	0.0447	0

Table 11. 5-gene estimation: trigonometric pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	34.6095	67.656	0.1118	0.0118	0.0479	5
Winbugs	44.9060	25.916	0.1129	0.0129	0.0506	0
REML	42.7539	0.294	0.1088	0.0088	0.0486	2
ML	36.5918	1.013	0.1089	0.0089	0.0468	0

Table 12. 5-gene estimation: exponential pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	27.5282	62.560	0.1190	0.0190	0.0428	6
Winbugs	28.8421	25.929	0.1167	0.0167	0.0426	2
REML	29.0559	0.323	0.1124	0.0124	0.0407	1
ML	28.8496	1.634	0.1131	0.0131	0.0407	0

Table 13. 5-gene estimation: interaction pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	52.170	53.809	0.096	-0.004	0.044	5
Winbugs	59.3075	26.819	0.0967	-0.0033	0.0457	1
REML	51.9351	0.318	0.0918	-0.0082	0.0423	3
ML	52.0457	0.880	0.0928	-0.0072	0.0425	0

In addition to simulating binary data for a 5-gene genetic pathway effect, we also simulated a binary response from a 10-gene pathway:

$$logit(\mu_i) = -5.0 + 0.1age_i + 0.10\sum_{j=1}^{10}\mathbf{Z}_{i,j} \tag{2.20}$$

$$\begin{aligned} logit(\mu_i) &= -5.0 + 0.1age_i + 0.1\mathbf{Z}_{i,1}^2 + 0.1\mathbf{Z}_{i,2}^2 + 0.15\mathbf{Z}_{i,3} + 0.1\mathbf{Z}_{i,4} + 0.25\mathbf{Z}_{i,5} + \\ &\quad 0.2\mathbf{Z}_{i,6}^2 + 0.15\mathbf{Z}_{i,7} + 0.1\mathbf{Z}_{i,8} + 0.05\mathbf{Z}_{i,9}^2 + 0.1\mathbf{Z}_{i,10} \end{aligned} \tag{2.21}$$

$$\begin{aligned} logit(\mu_i) &= -5.0 + 0.1age_i + 0.1\mathbf{Z}_{i,1}\mathbf{Z}_{i,2} + 0.15\mathbf{Z}_{i,3} + 0.1\mathbf{Z}_{i,4}\mathbf{Z}_{i,5} + \\ &\quad 0.2\mathbf{Z}_{i,6} + 0.15\mathbf{Z}_{i,7}\mathbf{Z}_{i,8} + 0.05\mathbf{Z}_{i,9} + 0.1\mathbf{Z}_{i,10} \end{aligned} \tag{2.22}$$

The above models represent genetic pathway effects that are linear, quadratic, and interaction respectively. The age covariate was generated from a discrete uniform from the ages of 35 to 55. The true value of the age coefficient is set to 0.1. A

100 simulated data sets were generated from these models with a sample size of 100 per simulation. The matrix, $\mathbf{Z}_{i,j}$, in the above models represent the gene expression measurement for sample i and gene j. The first 50 samples were generated from a multivariate normal with mean, $\boldsymbol{\mu} = \mathbf{0}$, and covariance structure of an AR-1 process with the AR parameter equaling 0.8. Samples 51 through 100 are generated from a multivariate normal with mean, $\boldsymbol{\mu} = (2.5, 2.5, 2.5, 2.5, 2.5, 0, 0, 0, 0, 0)$, and covariance structure of an AR-1 process with the AR parameter equaling 0.8.

To fit the 10-gene models, we again applied a logistic kernel machine, eq. 2.2, with a Gaussian kernel and kernel parameter set to 5. The following tables compare the ML approach based on the spherical radial approximation, the REML approach based on the spherical radial approximation, and the Gibbs sampling approach of Zeger et al [47]. The Gibbs sampler used 10000 iterations with a thinning parameter of 5 and a burnin period of 5000 iterations. As before, we compare the estimation performance by the SSR goodness of fit criterion.

Table 14. 10-gene estimation: linear pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	68.336	115.052	0.103	0.003	0.051	4
REML	69.806	2.149	0.095	-0.005	0.046	3
ML	66.639	4.439	0.103	0.003	0.049	2

Table 15. 10-gene estimation: quadratic pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	160.021	166.867	0.125	0.025	0.053	NA
REML	183.370	2.510	0.114	0.014	0.051	10
ML	175.554	6.359	0.129	0.029	0.065	7

Table 16. 10-gene estimation: interaction pathway effect

Method	SSR	Time (hrs)	âge	bias	sd(âge)	trap
Gibbs	92.835	159.924	0.115	0.015	0.049	NA
REML	153.058	1.664	0.125	0.025	0.054	6
ML	89.712	4.405	0.120	0.020	0.048	1

The above tables show that for a linear pathway effect, the ML's goodness of fit is better than Zeger's Gibbs sampling's goodness of fit. Additionally, ML has much lower computational cost than Gibbs sampling. For the quadratic pathway effect, the Gibbs result has a better goodness of fit compared to ML. Lastly, we looked at an interaction pathway effect, where our ML method is performing slighty better than Gibbs at a computational cost that is dramatically better than Gibbs.

2.5 Simulation study for testing of the genetic pathway effect

In this section, we performed a simulation study to compare the performance of our REML based likelihood ratio test with the score test proposed by D. Liu et al [29]. To this end, we generated testing data sets in the same manner as D. Liu. They considered linear and nonlinear genetic pathway effects, $h(\mathbf{z})$, and calculated the size and power for their proposed score test. Their true linear model is: $logit(\mu_i) = x + ah(\mathbf{z})$, where $h(\mathbf{z}) = 2z_1 + 3z_2 + z_3 + 2z_4 + z_5$. For their true nonlinear model, he considered: $logit(\mu_i) = x + ah(\mathbf{z})$, where $h(\mathbf{z}) = 2(z_1 - z_2)^2 + z_2z_3 + 3\sin(2z_3)z_4 + z_5^2 + 2\cos(z_4)z_5$. Z's were generated from a standard normal and $x = z_1 + \frac{e}{2}$ where e is generated from a standard normal independent of z_1. The 'a' values of 0, 0.2, 0.4, and 0.8 were considered, where $a = 0$ allows us to calculate the size of test and $a \neq 0$ allows us to compare the power between the REML

likelihood ratio test and the score test. For our power and size calculations, 100 simulations of sample size 100 were considered, and ρ was set to 5 for the REML LRT.

Table 17. 5-gene testing: nonlinear pathway results

Method	$a = 0$	$a = 0.2$	$a = 0.4$	$a = 0.8$
Score	0.13	0.15	0.47	0.56
REML LRT	0.02	0.37	0.72	0.94

Table 18. 5-gene testing: linear pathway results

Method	$a = 0$	$a = 0.2$	$a = 0.4$	$a = 0.8$
Score	0.21	0.71	1.00	1.00
REML LRT	0.05	0.47	0.93	1.00

For a nonlinear pathway effect, we see that D. Liu's size, 0.13, is inflated compared to .05, while our size is .02. The power of the REML LRT test is much greater than the score test for the 'a' values of 0.2, 0.4, and 0.8. As for the linear pathway effect, D. Liu's size is again inflated at .21 and our size, .05, matches the p-value cutoff. Our power is similar to D. Liu's for 'a' = 0.4 and 0.8, but we have smaller power for 'a' = 0.2.

2.6 Application to acute lymphocytic leukemia

As an application of our logistic kernel machine model, we applied our estimation and testing procedure to the analysis of an Acute Lymphocytic Leukemia data set from the Ritz Laboratory [27]. The data set consists of 12,625 gene expression measurements from 128 patients who have been diagnosed with B-cell ALL or T-cell ALL. 10,440 probe sets were able to be matched to at least one GO term in the biological process ontology and 4365 GO terms from the biological process

ontology were able to be mapped to one or more probe sets. The gene expression measurements have already been normalized by the Ritz laboratory using rma normalization. Additionally, the patient's gender, the patient's age, the date of diagnosis, the stage of the disease, the patient's ID, the patient's remission status, the patient's CR status, the date of remission, translocation detection status, cytogenetics status, molecular biology type, fusion protein status, and many other covariates were recorded.

From this data set, we would like to test for functional pathway differential expression comparing B-cell versus T-cell ALL. Liu et al [31] analyzed the ALL data set with their DEA-PLS and Fisher exact test approach and identified the top 10 differentially expressed categories (defined by significance) for each method. In the following tables, we compare our REML likelihood ratio testing approach with D. Liu's score test approach and Mielke and Berry's MRPP test [33] for testing these differentially expressed categories for differential expression.

Table 19. Testing for functional pathway differential expression

GOID	ngenes	p-value REML LRT	p-value Score Test	p-value MRPP
2263	13	0.006451613	5.935E-52	0
19883	6	0	0	0
19884	8	0	1.796E-28	0
19885	4	0	1.199E-46	0
19886	6	0	1.689E-28	0

Table 20. Testing for functional pathway non-differential expression

GOID	ngenes	p-value REML LRT	p-value Score Test	p-value MRPP
32387	5	0.741	0.668	0.970
51891	5	0.727	0.652	0.850
51890	5	0.727	0.652	0.850
60038	5	0.727	0.652	0.820
31018	5	0.731	0.613	0.740

All three of the testing procedures show that all of the functional categories are differentially expressed. Here, the score test and the likelihood ratio test adjusts for the age covariate, but MRPP does not adjust for age.

Next, we investigated functional categories that are non-differentially expressed. 304 GO categories out of the 4365 categories consisted of 5 genes or 10 genes. These 304 categories were extracted and tested by the MRPP test. The largest 5 p-values of the 304 p-values were identified, and they had values of 0.740, 0.824, 0.848, 0.848, and 0.968. Consequently, the categories associated with these p-values are non-differentially expressed. These non-differentially expressed categories have the GO IDs of 31018, 60038, 51890, 51891, and 32387. We then tested these five categories using the score test and the REML likelihood ratio test. The table above shows that all three methods are in agreement that the five categories are non-differentially expressed.

In addition to testing for differential expression and non-differential expression, we would like to apply our estimation procedure to the ALL data set. Here, we focused on estimation of the age effect for the GO categories of GOID: 2263, GOID: 19883, GOID: 19884, GOID: 19885, and GOID: 19886. Estimates of the age effect were obtained by maximum likelihood and REML. We also obtained 95 percent confidence intervals for the age effect by parametric bootstrapping.

Table 21. Confidence intervals for age for GOID: 2263

Method	$\hat{age}$	2.5Q	97.5 Q	sd	ngenes	converged?
ML-ALL	-0.0454	-1.6522	-0.0330	0.402	13	426/497
REML-ALL	-0.0310	-0.8013	0.0372	0.220	13	184/200

Table 22. Confidence intervals for age GOID: 19883

Method	$\hat{age}$	2.5Q	97.5 Q	sd	ngenes	converged?
ML-ALL	-0.0560	-1.6973	0.2273	0.486	6	140/200
REML-ALL	0.002	-0.553	0.522	0.297	6	169/200

Table 23. Confidence intervals for age for GOID: 19884

Method	$\hat{age}$	2.5Q	97.5 Q	sd	ngenes	converged?
ML-ALL	-0.002	-0.937	0.525	0.364	8	157/200
REML-ALL	-0.0039	-0.7232	0.3338	0.279	8	160/200

Table 24. Confidence intervals for age for GOID: 19885

Method	$\hat{age}$	2.5Q	97.5 Q	sd	ngenes	converged?
ML-ALL	-0.0175	-0.8605	0.2308	0.299	4	197/200
REML-ALL	-0.0156	-0.7267	0.2096	0.244	4	188/200

Table 25. Confidence intervals for age for GOID: 19886

Method	$\hat{age}$	2.5Q	97.5 Q	sd	ngenes	converged?
ML-ALL	-0.0076	-0.9574	0.4254	0.331	6	184/200
REML-ALL	-0.0010	-0.7163	0.4501	0.297	6	176/200

The above tables illustrate that the REML method has shorter confidence intervals and smaller standard errors compared to ML. Furthermore, age should not be included in the model building for the functional pathways with GOIDs: 19883, 19884, 19885, and 19886 since 0 is contained in the confidence interval.

2.7 Future directions

In the future, we hope to extend our single pathway model to handle the effect of multiple gene sets/pathways on a binary outcome. To this end, we could employ ideas such as the backfitting method [17], Luan and Li's boosting method [32], Truccolo's boosting method [43], Genkin's coordinate descent [11], or Friedman's boosting method [10]. Additionally, we hope to generalize our methods to handle other types of response variables such as exponential, gamma, or Poisson.

BIBLIOGRAPHY

[1] Booth, J. and Hobert, J. (1999). Maximizing generalized linear mixed model likelihoods with an automated Monte Carlo EM algorithm. *Journal of the Royal Statistical Society, Series B*, **62**, 265-285.

[2] Breslow, N. E. and Lin, X. (1995). Bias correction in generalized linear mixed models with a single component of dispersion. *Biometrika*, **82**, 81-91.

[3] Chen, J., Zhang D., et al. (2002). A Monte Carlo EM algorithm for generalized linear mixed models with flexible random-effects distribution. *Biostatistics*, **3**, 347-360.

[4] Chib, S. and Jeliazkov, I. (2006). Inference in semiparametric dynamic models for binary longitudinal data. *Journal of the American Statistical Association*, **101**, 685-700.

[5] Breslow, N.E. and Clayton, D.G. (1993). Approximate Inference in Generalized Linear Mixed Models, *Journal of the American Statistical Association*, **88**, 9-25.

[6] Clarkson, D. and Zhan, Y. (2002). Using Spherical Radial Quadrature to Fit Generalized Linear Mixed Effects Models. *Journal of Computational and Graphical Statistics*, **11**, no. 3, 639-659.

[7] Crainiceanu, C. M., Ruppert, D. and Wand, M. (2005). Bayesian analysis for penalized spline regression using winbugs. *Journal of Statistical Software*, **14(14)**.

[8] Efron B. (2001). Selection criteria for scatterplot smoothers. *Annals of Statistics*, **29**, 470-504.

[9] Efron, B. and Tibshirani, R. (2006). On testing the significance of sets of genes. Tech Report.

[10] Friedman, J. (1999). Greedy Function Approximation: A gradient boosting machine. Salford Systems.

[11] Genkin, A., Lewis, D., et al. (2007). Large Scale Bayesian Logistic Regression for Text Categorization. *Technometrics*, **49**, 3, 291-304.

[12] Gibbons, R. and Hedeker D. (1997). Random effects probit and logistic regression models for three level data. *Biometrics*, **53**, 1527-1537.

[13] Goeman, J.J. and Buhlmann, P. (2007). Analyzing gene expression data in terms of gene sets: methodological issues. *Bioinformatics*, **23**, 980-987.

[14] Goeman, J., Geer, S., et al. (2004). A global test for groups of genes: testing association with a clinical outcome. *Bio-infomatics*, **20**, no. 1, 93-99.

[15] Goldstein, H. I. (1991). Nonlinear multilevel models, with an application to discrete response data. *Biometrika*, **78**, 45-51.

[16] Gu, C. and Ma, P. (2005). Generalized Nonparametric Mixed-Effect Models: Computation and Smoothing Parameter Selection. *Journal of Computational and Graphical Statistics*, **14**, 485-504.

[17] Hardle, W., Muller, M., et al. Nonparametric and Semiparametric Models - An Introduction. e-book. http://fedc.wiwi.hu-berlin.de/xplore/ebooks/html/spm/spmhtmlframe126.html

[18] Hedeker, D. (1999). Mixno: a computer program for mixed-effects nominal logistic regression. *Journal of Statistical Software*, **4(5)**.

[19] Hedeker, D. and Gibbons, R. (1996). MIXOR: A computer program for mixed-effects ordinal regression analysis. *Computer Methods and Programs in Biomedicine*, **49**, 157-176.

[20] Karcher, P. and Wang, Y. (2001). Generalized nonparametric mixed effects models. *Journal of Computational and Graphical Statistics*, **10**, 641-655.

[21] Kuk, A. Y. C. (1995) Asymptotically unbiased estimation in generalized linear models with random effects. *J. R. Statist. Soc. B*, **57**, 395-407.

[22] Liang, H. (2007). Generalized partially linear mixed-effects models incorporating mismeasured covariates. *Annals of the Institute of Statistical Mathematics*, **61**, No. 1, 27-46.

[23] Lee, K. J. and Thompson S. G. (2007). Flexible parametric models for random effects distributions. *Statistics in Medicine*, DOI:10.1002/sim.2897.

[24] Litière, S., Alonso A., and Molenberghs G. (2008). The impact of a misspecified random-effects distribution on the estimation and performance of inferential procedures in generalized linear mixed models. *Statistics in Medicine*, **27**, 3125-3144.

[25] Longford, N.T. (1988). A quasi-likelihood adaptation for variance component analysis. *Proc. Sect. Comp. Statist., Am. Statist. Assoc.*

[26] Longford, N. T. (1994). Logistic regression with random coefficients. *J. Comput. Statist. Data Anal.*, **17**, 1-15.

[27] Li, X. ALL data package - data of T and B-cell Acute Lymphocytic Leukemia from the Ritz Laboratory. http://www.bioconductor.org/packages/release/data/experiment/html/ALL.html

[28] Lin, X. and Zhang, D. (1999). Inference in generalized additive mixed models by using smoothing splines. *Journal of the Royal Statistical Society, Series B*, **61**, 381-400.

[29] Liu, D., Ghosh, D., et al. (2008). Estimation and testing for the effect of a genetic pathway on a disease outcome using logistic kernel machine regression via logistic mixed models. *BCM Bio-infomatics*, **9**: 292.

[30] Liu, D., Lin, X., and Ghosh, D. (2007). Semiparametric Regression of Multidimensional Genetic Pathway Data: Least Squares Kernel Machines and Linear Mixed Models. *Biometrics*, **63**, 1079-1088.

[31] Liu, J., et al. (2007). Domain enhanced analysis of microarray data using GO annotations. *Bioinformatics*, **23**, 1225-1234.

[32] Luan, Y. and Li, H. (2006). Group Additive Regression Models for Genomic Data Analysis. UPenn Biostatistics Working Papers, University of Pennsylvania.

[33] Mielke, P. and Berry, K. (2001) Permutation methods: A Distance Function Approach. Springer-Verlag. New York.

[34] Monahan, J. and Genz, A. (1996). Spherical Radial Integration Rules for Bayesian Computation. *Journal of the American Statistician Association*, **92**, 664-674.

[35] Nettleton, D., Recknor, J., and Reecy, J. (2008). Identification of differential expression gene categories in microarray studies using nonparametric multivariate analysis. *Bio-infomatics*, **24**, no. 2, 192-201.

[36] Noh M. and Lee Y. (2007). REML estimation for binary data in GLMMs. *Journal of Multivariate Analysis*, **98**, 896-915.

[37] Pinheiro, J.C. and Chao, E.C. (2006). Efficient Laplacian and Adaptive Gaussian Quadrature Algorithms for Multilevel Generalized Linear Mixed Models, *Journal of Computational and Graphical Statistics*, **15**, no. 1, 58-81.

[38] Raudenbush, S., Bryk, A., Cheong, Y. and Congdon, R. (2000). *HLM 5: Hierarchical linear and nonlinear modeling. (Statistical software manual).* Scientific Software International, Lincolnwood, IL.

[39] Rodriguez, G. and Goldman, N. (1995). An Assessment of Estimation Procedures for Multilevel Models with Binary Responses, *Journal of the Royal Statistical Society Series A*, **158**, 73-89.

[40] Rodriguez, G. and Goldman, N. (2001). Improved estimation procedures for multilevel models with binary response: case study, *J. R. Statist. Soc. A*, **164**, Part 2, 339-355.

[41] Ruppert, D., Wand, M. P. and Carroll, R. J. (2003). *Semiparametric Regression*, Cambridge University Press.

[42] Subramanian, A., Tamayo, P., et al. (2005). Gene set enrichment analysis: A knowledge based approach for interpreting genome-wide expression profiles. *PNAS*, **102**, no. 43, 15545-15550.

[43] Truccolo, W. and Donoghue, J. (2007). Nonparametric Modeling of Neural Point Processes via Stochastic Gradient Boosting Regression. *Neural Computation*, **19**, 672-705.

[44] Wahba, G. (1990). *Spline models for observational data*. CBMS-NSF Regional Conference Series in Applied Mathematics, SIAM.

[45] Wand, M.P. (2003). Smoothing and mixed models. *Computational Statistics*, **18**, 223-249.

[46] Young-Ju, K. and Gu, C. (2004). Smoothing Spline Gaussian Regression: More Scalable Computation via Efficient Approximation. *Journal of Royal Statistical Society B*, **66**, 337-356.

[47] Zeger, S. and Karim, M. (1991). Generalized Linear Models with Random Effects: A Gibbs sampling approach. *Journal of the American Statistical Association*, **86**, No. 413, 79-86.

[48] Zhao, Y., Staudenmayer, J., Coull, B. and Wand, M. (2006). General design Bayesian generalized linear mixed models. *Statistical Science*, **21**, 35-51.