ABSTRACT

SCHMIDT, KATHLEEN LYNN. Uncertainty Quantification for Mixed-Effects Models with Applications in Nuclear Engineering. (Under the direction of Dr. Ralph C. Smith.)

Mixed-effects models include two types of parameters: fixed effects, which characterize the nominal parameter value for a population, and random effects, which characterize the variation among individual data sets. Whereas this type of model is routinely used in a variety of scientific fields, there has been little consideration for quantifying the associated uncertainties. In this dissertation, we explore techniques for performing uncertainty quantification (UQ) on mixed-effects models, focusing on the tasks of model calibration and parameter selection.

To aid in model calibration, we introduce a novel version of the Delayed Rejection Adaptive Metropolis (DRAM) algorithm for mixed-effects models. Moreover, we employ this new technique to calibrate nuclear engineering models, including a parameterized version of the Dittus-Boelter model. We also utilize the modified DRAM algorithm for radiation source localization in an urban setting based on detector responses. We consider this inverse problem for both stationary and mobile detectors, and we incorporate mixed-effects modeling to account for the variation in background radiation among detector locations.

The parameterizations of mixed-effects models that serve to incorporate the population and individual effects are often unidentifiable in the sense that parameters are not uniquely specified by the data, but traditional parameter selection techniques are ineffective. As a result, current literature focuses on model selection, by which insensitive parameters are fixed or removed from the model. Model selection methods that employ information criteria are applicable to both linear and nonlinear mixed effects models, but such techniques are limited in that they are computationally prohibitive for large problems due to the number of possible models that must be tested. To limit the scope of possible models for model selection via information criteria, we introduce a parameter subset selection (PSS) algorithm for mixed-effects models, which orders the parameters by their significance. We provide examples to verify the effectiveness of the PSS algorithm and to test the performance of mixed-effects model selection that makes use of parameter subset selection.

Uncertainty Quantification for Mixed-Effects Models with Applications in Nuclear Engineering

by
Kathleen Lynn Schmidt

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Applied Mathematics

Raleigh, North Carolina

2016

APPROVED BY:

Dr. Alen Alexanderian	Dr. John Mattingly
Dr. Hien Tran	Dr. Ralph C. Smith Chair of Advisory Committee

BIOGRAPHY

Kathleen grew up in the suburbs of Kansas City, MO and stayed in-state to attend Missouri State University as an undergraduate. After graduating *summa cumme laude* with a B.A. in Mathematics in 2007, Kathleen returned to Missouri State to as a graduate student. In 2009, she completed her M.S. in Natural and Applied Science and moved to Hot Springs, AR to teach at the Arkansas School of Mathematics, Sciences, and the Arts (ASMSA). After three years of teaching, two at ASMSA and one as an instructor at Missouri State, Kathleen entered the Applied Mathematics Ph.D. program at North Carolina State University. After completing her thesis defense, Kathleen will join the Applied Statistics group at Lawrence Livermore National Lab as a postdoctoral researcher.

ACKNOWLEDGEMENTS

Thank you to my advisor Dr. Ralph Smith for his support, encouragement, and input. Over the last few years, I have become a better mathematician, and I'm extremely grateful for the opportunities that I've had under your advisement. Thanks to you, I'm ready to take on the mathematical world! Also, thank you to the other members of my committee—Dr. Alen Alexanderian, Dr. John Mattingly, and Dr. Hien Tran—for their time and their interest in my work.

To all of my officemates and math friends, thank you for the support and sense of community during my time at NCSU. Special thanks to Allie Lewis for helping to beta test The Buddy System™. You helped celebrate my successes on the good days and let me pet your cats on the bad days. I could not have done it without you. Or the cats.

Thank you to all of my family for taking such pride in my academic achievements. Special thanks to my parents and my sister, Jackie, for always believing in me. Also, thanks to all of my long-distance friends for encouraging me from afar.

This research was funded by:

- DOE Consortium for Advanced Simulation of LWR (CASL)
- NSF Grant CMMI-1306290
- NNSA Consortium for Nonproliferation Enabling Capabilities (CNEC)

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER

1

INTRODUCTION

Uncertainty quantification (UQ) is the science of identifying and reducing sources of uncertainty in order to make predictions and understand the degree to which these predictions can be trusted. The field of UQ is inherently multidisciplinary, incorporating aspects such as mathematical modeling, statistics, and numerical analysis. As shown in Figure 1.1, model calibration and parameter selection are vital aspects of UQ. Model calibration generally serves as an initial step in quantifying uncertainties. Parameter selection—typically implemented via sensitivity analysis or active subspace construction—isolates a subset or subspace of influential and identifiable parameters. This aids model calibration by reducing the number of parameters to be estimated and ensuring that there exists a unique set of inferred parameters.

1.1 Model Calibration

Model calibration involves optimally inferring parameters to match the model output to a physical response obtained from measurement data. For the purposes of UQ, we also want to quantify, or possibly update, the uncertainty in these optimal parameter estimates. We can accomplish this either by constructing parameter distributions or by determining confidence intervals about a parameter estimate. We examine two perspectives on parameter estimation: frequentist and Bayesian.

From a frequentist point of view, probabilities are defined as the frequency with which events

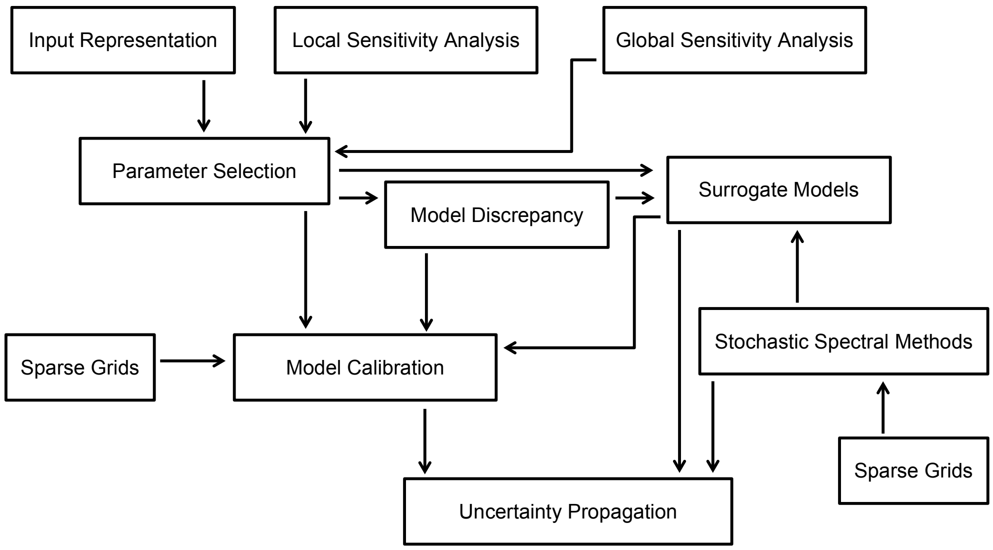


Figure 1.1: Flow chart representing the components of predictive estimation in uncertainty quantification as described in [37].

occur when a large number of experiments are performed. Thus, a frequentist views the concept of probability as a deterministic value that does not change regardless of experimental data. Similarly, parameters are also viewed as fixed but unknown values, which are not affected by the collection of additional response data. To estimate these fixed parameter values, we construct estimators for assigning optimal parameter values based on response data. These estimators are functions of random variables that map the sample space—that is, the set of all possible observations—to a set of parameter estimates. Hence, the estimators themselves are considered random variables, each with an associated sampling distribution. Since the parameters are assumed to have a true, fixed value, parameter uncertainty from a frequentist perspective is simply the uncertainty of the estimator, which is represented by its sampling distribution [37].

The Bayesian perspective defines probability as a quantified measure of the belief that an event will occur based on available information and prior knowledge [1]. Note that this interpretation of probability is subjective; hence, Bayesian probabilities are not fixed values and can change as more information is acquired. As another departure from the frequentist framework, Bayesian parameters are regarded as random variables; their associated distributions characterize the current "state of knowledge" about the parameter value. Hence, the task of Bayesian parameter estimation involves constructing the parameter probability density function (pdf), which is termed the "posterior density," rather than obtaining a single-valued approximation. Since Bayesian probability is conditioned on observations and prior information, Bayes' theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

for events A and B where $P(*)$ denotes the probability of an event occurring provides a natural foundation for parameter estimation. Thus, to infer parameters $Q = [Q_1, Q_2, \ldots, Q_p]$ based on observations $\nu = [\nu_1, \nu_2, \ldots, \nu_n]$, we employ Bayes' relation

$$\pi(q|\nu) = \frac{\pi(\nu|q)\pi_0(q)}{\int_{\mathbb{R}^p} \pi(\nu|q)\pi_0(q)dq}, \tag{1.1}$$

where $\pi(q|\nu)$ is the posterior parameter pdf, q represents realizations of Q, $\pi_0(q)$ is the prior distribution, $\pi(q|\nu)$ is the likelihood function, and the marginal pdf represented by the integral in the denominator is a normalization factor [37]. While (1.1) appears to be a straightforward formula for obtaining the posterior density, its implementation can be difficult in practice. The normalization factor in the denominator can rarely be calculated analytically, so numerical methods such as quadrature techniques must instead be applied. This is similarly true for the integral evaluations required to obtain marginal posterior densities from the joint posterior $\pi(q|\nu)$. As an alternative to numerically evaluating theses integrals, we can construct Markov chains whose stationary distribution is the posterior density as is done in Markov Chain Monte Carlo (MCMC) techniques [37]. In Chapter 2, we construct distributions using both frequentist and Bayesian techniques to explore parameter uncertainty.

1.2 Sensitivity Analysis

Sensitivity analysis involves quantifying the relative contributions of parameters or inputs to the model output [37]. One important application of sensitivity analysis is parameter selection. Once insensitive parameters are identified, they can be fixed rather than estimated with minimal impact on the model response. This is particularly beneficial for models in biology and physics, which often have hundreds of parameters [45]. Reducing the number of parameters, especially in high-dimensional problems, greatly improves the efficiency—and sometimes the feasibility—of model calibration.

Sensitivity analysis techniques are divided into two categories: local and global. Local sensitivity analysis methods examine how the model response varies when the parameters or inputs are perturbed about a nominal value. Partial derivatives are typically employed to quantify local sensitivities, but they are often impossible or infeasible to calculate directly. Whereas adjoint capabilities are available for certain codes, they are not generally available for the thermal-hydraulics and fuel codes employed for motivating CASL applications summarized in Section 1.4.1. Common techniques for obtaining local sensitivities include finite difference approximations, solutions to sensitivity equations, and automatic differentiation [37]. Whereas the majority of sensitivity analysis literature focuses on local techniques, such methods can be problematic for determining the global effect of parameters, especially in highly nonlinear problems [33, 37]. When the sensitivity over the entire parameter space is of interest, global techniques are advantageous.

Global sensitivity techniques ascertain the relative contributions of parameter uncertainty to the uncertainty in the model output over the entire possible range of parameter values. Such global sensitivities depend solely on the model and response and are not affected by experimental data [37]. Variance-based global sensitivity methods, such as the calculation of Sobol' indices, apportion the variance of the output $\mathrm{Var}(y)$ to the variance of the parameters. To do this, we rank the parameters $q = [q_1, q_2, \ldots, q_p]$ based on the amount of variance that is removed from the output when a particular parameter is fixed. Ideally, we would fix the parameters, setting them equal to nominal values q_i^*, and calculate $\mathrm{Var}(Y|q_i = q_i^*)$ for each parameter q_i, but these values of q_i^* are generally unknown. We instead take the average of the variance over all possible values of q_i, namely $\mathbb{E}\big(\mathrm{Var}(y|q_i)\big)$ [34]. Although some would recommend using variance-based methods whenever possible [33], these sensitivity methods are computationally demanding, which often makes them infeasible for complex and high-dimensional problems. In such cases, Morris screening provides an appealing alternative. The idea of Morris screening is to average local sensitivity information, essentially finite difference approximations of partial derivatives, taken throughout the parameter space to obtain a more global measure of sensitivity. Unlike variance-based methods, Morris screening only provides a relative ranking of parameter significance; it does not give a measure of how much more significant a higher-ranking parameter is [37]. In spite of providing less information, Morris screening remains a popular choice for global sensitivity analysis due to its computational efficiency.

1.3 Uncertainty Quantification for Mixed-Effects Models

Mixed-effects models are commonly used to statistically model phenomena that include attributes associated with a population or general underlying mechanism as well as effects specific to individuals or components of the general mechanism. This can include individual effects associated with data from multiple experiments. When appropriate, the incorporation of mixed-effects can reduce model discrepancy and provide a means for quantifying individual variation of parameter values within populations.

Despite the advantages of using this framework, uncertainty quantification for mixed-effects models is particularly challenging since UQ techniques established for traditional modeling generally prove incompatible or ineffective with this type of model. In this dissertation, we focus on parameter estimation and sensitivity analysis methods for mixed-effects models. In Chapter 3, we detail the current procedures—both Bayesian and frequentist—for mixed-effects parameter estimation and introduce a modified version of the Delayed Rejection Adaptive Metropolis (DRAM) algorithm for mixed-effects models. Current frequentist methods for mixed-effects parameter estimation, which involve on maximum likelihood estimation, are available in MATLAB via the Statistics Toolbox. The standard Bayesian technique for mixed-effects models is Gibbs sampling, which is also utilizing for some parameter updates in our modified DRAM algorithm. In Chapter 4, we demonstrate the problems with applying traditional sensitivity analysis techniques to mixed-effects models and propose an efficient method for mixed-effects parameter selection that is effective for both linear and nonlinear problems. While traditional sensitivity analysis techniques fail to distinguish between the global parameters and the parameters quantifying individual variations, our parameter subset selection algorithm, based on standard errors, accurately ranks both types of parameters for mixed-effects models.

1.4 Applications

Mixed-effects models have applications in many areas of science and engineering. We specifically explore nuclear engineering applications, focusing on problems that are of interest to the **C**onsortium for **A**dvanced **S**imulation of **L**ight-water Reactors (CASL) and to the **C**onsortium for **N**onproliferation **E**nabling **C**apabilities (CNEC).

1.4.1 CASL Applications

CASL was founded with the purpose of improving modeling and simulation for the light-water reactor (LWR). Unlike heavy water reactors used in Canada and India, light-water nuclear reactors employ ordinary water as a coolant and neutron moderator [37]. With the aim of modeling this reactor type, CASL created the **V**irtual **E**nvironment for **R**eactor **A**pplications (VERA). This environment includes capabilities for thermal-hydraulics analysis, which is crucial for modeling the behavior of the coolant. The coolant in the LWR is present in both liquid and vapor form; hence, we require a two-phase model.

Let α_g and α_f represent the volume fractions for the gas and fluid phases. We respectively denote the densities and velocities of the gas and fluid phases as ρ_g, ρ_f and v_g, v_f. Let the internal energies of gas and fluid be denoted by e_g and e_f . Now, using conservation of mass, momentum, and energy, we can model the fluid phase relations as

$$\frac{\partial}{\partial t}(\alpha_f \rho_f) + \nabla \cdot (\alpha_f \rho_f v_f) = -\Gamma,$$

$$\alpha_f \rho_f \frac{\partial v_f}{\partial t} + \alpha_f \rho_f v_f \cdot \nabla v_f + \nabla \cdot \sigma_f^R + \alpha_f \nabla \cdot \sigma + \alpha_f \nabla \rho_f$$
$$= -F^R - F + \Gamma(v_f - v_g)/2 + \alpha_f \rho_f g,$$

and

$$\frac{\partial}{\partial t}(\alpha_f \rho_f e_f) + \nabla \cdot (\alpha_f \rho_f e_f v_f + Th) = (T_g - T_f)H + T_f \Delta_f$$
$$- T_g(H - \alpha_g \nabla \cdot h) + h \cdot \nabla T - \Gamma[e_f + T_f(s^* - s_f)]$$
$$- p_f \left(\frac{\partial \alpha_f}{\partial t} + \nabla \cdot (\alpha_f v_f) + \frac{\Gamma}{\rho_f} \right),$$

where T_f is the fluid temperature, s_f is the fluid entropy density, p_f is the continuous phase pressure, σ is the viscous transport coefficient, and κ, ζ, and γ are positive transport coefficients [37]. The coupled relations for the gas phase are analogous. In addition to these equations, numerous closure relations, such as the Dittus-Boleter equation, are needed to model the coolant. In Chapter 2, we introduce a parameterized version of the phenomenological Dittus-Boelter equation. We then construct pdf's for the parameters and illustrate the need for model modifications, including the incorporation of mixed-effects.

1.4.2 CNEC Applications

CNEC, funded by a grant from the National Nuclear Security Administration (NNSA), is comprised of seven universities (North Carolina State University, Georgia Institute of Technology, Kansas State University, North Carolina A&T State University, Purdue University, University of Illinois at Urbana-Champaign, and University of Michigan) and three national laboratories (Los Alamos, Oak Ridge, and Pacific Northwest National Laboratories). This consortium supports research in the detection and characterization of special nuclear materials (SNM) as well as in the detection of facilities producing SNM. CNEC members also investigate feasible replacements for industrial radiation sources as a means to prevent their misappropriation such as being used to build dirty bombs.

In accordance with the goals of CNEC, we investigate radiation detection in an urban setting in Chapters 5 and 6. Given responses from radiation detectors, we wish to determine the radiation source intensity and location. In Chapter 5, we solve this inverse problem for stationary detectors using a simplified radiation transport model. However, this model does not account for variation in background radiation among the detector locations. We introduce a mixed-effects model in Chapter 6 to account for the varying background term. In addition to stationary radiation sensors, we also explore mobile detectors. In Chapter 5, we propose an algorithm to guide the movement of mobile sensors using mutual information to determine the location that provides the most information.

1.5 Dissertation Contributions and Organization

In this dissertation, we introduce two new UQ techniques for mixed-effects models: a mixed-effects version of the DRAM algorithm and a parameter subset selection (PSS) algorithm. The DRAM algorithm for mixed-effects models provides a new method of Bayesian parameter estimation, and the PSS algorithm aids mixed-effects model selection when traditional sensitivity analysis techniques are ineffective. Moreover, we employ mixed-effects modeling for a variety of nuclear engineering problems, including radiation detection in an urban setting. The organization of this dissertation, based on the contents of the chapters, is detailed below.

- **Chapter 2**
  We introduce a parameterized version of the Dittus-Boelter equation, which serves as a motivating example for the use of mixed-effects modeling. As mentioned in Section 1.4.1, the Dittus-Boelter equation is important to the CASL initiative because it serves as one of the closure relations in the LWR coolant model. We construct parameter pdf's for the parameterized Dittus-Boelter equation using three methods: asymptotic analysis, bootstrapping, and DRAM. Also, we provide a plot of experimental data that suggests that the Dittus-Boelter model parameters are inconsistent among data sets, indicating that use of mixed-effects modeling is advisable.

- **Chapter 3**
  We formally introduce the structure of mixed-effects models and highlight the parameter estimation techniques for such models. Using functions from the MATLAB Statistics Toolbox, we perform frequentist parameter estimation, via maximum likelihood estimation, for both linear and nonlinear mixed-effects models. For Bayesian parameter estimation, Gibbs sampling is the current standard, but some efforts have been made to expand the DRAM algorithm to mixed-effects models. In particular, the MATLAB MCMC Toolbox DRAM code contains an option for estimating mixed-effects parameters with independent random effects. We expand upon this and introduce a novel version of the DRAM algorithm for mixed-effects models with non-diagonal random effects covariance matrices.

- **Chapter 4**
  When performing sensitivity analysis for mixed-effects models, traditional techniques are generally ineffective, failing to distinguish between the global and local effects. In the mixed-effects literature, the isolation of sensitive parameters is instead achieved with model selection via information criteria. However, this process can be computationally prohibitive, especially for high-dimensional problems. Alternatives to this type of model selection have been proposed, but most of these cannot be applied to nonlinear mixed-effects models. To remedy these problems, we introduce a novel parameter subset selection algorithm for mixed-effects

models, which is applicable to both linear and nonlinear problems, and use it to reduce the computational cost of model selection with information criteria.

- **Chapter 5**
  Here we introduce the problem of radiation source localization. We first consider the case of stationary radiation detectors. To simulate a radiation source in downtown Washington D.C., we use a simplified photon transport equation to generate the detector responses. Using this data, we solve the inverse problem to determine the intensity and location of the source. In this chapter, we do not employ mixed-effects for our detector response model; we simply employ the photon transport equation used to generate the data as our model. We perform parameter estimation via DRAM and the Differential Evolution Adaptive Metropolis (DREAM) algorithm. We also consider source localization with mobile sensors. We propose the use of mutual information to guide the movement of the radiation detectors. In particular, we provide an algorithm that determines the optimal measurement location from a set of possible design conditions using mutual information.

- **Chapter 6**
  The simplified photon transport model from Chapter 5 does not account for varying background radiation among the detector locations. Thus, we incorporate mixed-effects to allow for individual background We apply the mixed-effects DRAM algorithm form Chapter 3 to estimate the source location and intensity along with the individual background parameters for each source. We initially used flat priors for all of the parameters and discovered that the parameter set is not mutually identifiable. We use the term “flat prior" rather than "uniform prior" throughout this disseration because the employed prior distributions may not integrate to unity and may, in fact, be improper; e.g. for parameters with the admissible space $[0,\infty)$. We then calibrated the background radiation parameters in the absence of a source to obtain better prior information for the background terms. Use of this narrow prior distributions allowed us to simultaneously estimate all of the model parameters without experiencing identifiability issues.

CHAPTER

2

STATISTICAL INFERENCE FOR THE DITTUS-BOELTER EQUATION

The Dittus-Boelter equation for heated liquids is

$$Nu = 0.023 Re^{0.8} Pr^{0.4}, \tag{2.1}$$

where Nu is the Nusselt number, Re is the Reynolds number, and Pr is the Prandtl number with Pr having an exponent of 0.3 for cooling liquids. This empirical relation is frequently used for approximate calculations in engineering. Pinpointing the origin of the equation in its final form is somewhat challenging with many authors referencing papers that do not contain the equation at all [47]. The most likely true origin is McAdams's 1942 textbook [26] with the author slightly modifying the values of the coefficient and exponents compared to earlier versions of the equation.

The Dittus-Boelter equation was first developed to describe heat transfer in the smooth pipes of automobiles, but it is currently employed by CASL to describe heat transfer in light-water nuclear reactors [28]. Currently, CASL utilizes the thermal hydraulic code CTF—originally called COBRA-TF before its rebranding—as a component of its virtual reactor (VERA) [32]. CTF uses the Dittus-Boelter equation to model heat transfer from the solid wall to fluids of certain regimes in the reactor pipes. In particular, the Dittus-Boelter equation is used for fluids that are categorized as single-phase vapor or turbulent single-phase liquids. For saturate nucleate boiling, CTF employs Chen and Thom correlations, which are modified versions of the Dittus-Boelter equation [28].

The nominal values of 0.023, 0.8, and 0.4 from (2.1) were obtained via fitting the data by hand for a specific regime. Since the Dittus-Boelter has taken various forms [47] due to varying calibration regimes and the use of rudimentary parameter estimation methods, we utilize a parameterized version to obtain more accurate coefficient and exponent values via parameter estimation. Thus, for modeling the behavior of heated liquids, we consider the statistical model

$$Nu(q, Re, Pr) = q_1 Re^{q_2} Pr^{q_3} + \varepsilon, \tag{2.2}$$

where Nu is the Nusselt number, Re is the Reynolds number, Pr is the Prandtl number, and ε is the measurement error. We assume that the measurement errors are independent and identically distributed (iid); in particular, $\varepsilon \sim \mathcal{N}(0, \sigma^2)$. Note that (2.2) is a parameterized version of the Dittus-Boelter equation (2.1).

2.1 Parameter Probability Density Functions

Whereas we seek to calibrate the model (2.2) via parameter estimation, we also aim to quantify the uncertainty of these parameter estimates. We do this by constructing parameter probability density functions (pdf's). To implement parameter estimation and uncertainty quantification, we employed experimental data sets from [27], namely groups of ordered triples containing recorded measurements of the Reynolds, Prandtl, and Nusselt numbers. We utilized four such groups corresponding to four different steam-heated liquids: water, gas oil, straw oil, and light motor oil. Given the four data sets—which respectively contained 12, 13, 22, and 9 data points—we constructed pdf's for the parameter set $q = [q_1, q_2, q_3]$ using three methods: asymptotic analysis, bootstrapping, and the Delayed Rejection Adaptive Metropolis (DRAM) algorithm.

2.1.1 Asymptotic Analysis

As explained in Chapter 1, the frequentist approach to quantifying parameter uncertainty involves examining the sampling distribution, which characterizes the uncertainty in the performance of the estimator. For asymptotic analysis, we consider the behavior of the sampling distribution as $n \to \infty$ where n is the number of data points. Here, we examine the asymptotic behavior of the sampling distribution associated with the nonlinear ordinary least squares (OLS) estimator.

Let a general nonlinear statistical model be represented by

$$\Upsilon = f(q_0) + \varepsilon$$

where Υ is the random vector of measurements, f is the mathematical model function, q_0 represents the true but unknown frequentist parameter vector, and $\varepsilon = [\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_n]^T$ is the vector of measurement errors. As with the Dittus-Boelter model (2.2), we assume that the errors are iid and normally distributed with a mean of zero and fixed but unknown variance, which we denote here as σ_0^2. The nonlinear OLS estimator and estimate are defined as

$$\begin{aligned} q_{OLS} &= \underset{q\in\mathscr{Q}}{\operatorname{argmin}} \sum_{i=1}^{n} \left[\Upsilon_i - f_i(q)\right]^2, \\ \hat{q}_{OLS} &= \underset{q\in\mathbb{Q}}{\operatorname{argmin}} \sum_{i=1}^{n} \left[\upsilon_i - f_i(q)\right]^2, \end{aligned} \tag{2.3}$$

where $\mathscr{Q}$ is the space associated with the estimator q_{OLS}, $\mathbb{Q}$ is the admissible parameter space, and $\upsilon = [\upsilon_1, \upsilon_2, \ldots, \upsilon_n]$ is the vector of experimental observations. Hence, the parameter estimate $\hat{q}_{OLS}$ is obtained from minimizing the cost function

$$\mathscr{J}(q) = \sum_{i=1}^{n} \left[\upsilon_i - f_i(q)\right]^2 \tag{2.4}$$

subject to $q \in \mathbb{Q}$. For nonlinear problems, (2.4) generally does not have an analytic solution, so we instead minimize the cost function numerically. In this dissertation, we employ the MATLAB function `fminsearch` to obtain nonlinear OLS estimates. Since the error variance is also fixed but unknown, we also construct an OLS estimator and estimate

$$\begin{aligned} \sigma_{OLS}^2 &= \frac{1}{n-p} R^T R \\ \hat{\sigma}_{OLS}^2 &= \frac{1}{n-p} \hat{R}^T \hat{R} \end{aligned}$$

for the error variance where p is the number of parameters. Here, $R = \Upsilon - f(q_{OLS})$ and $R = \upsilon - f(\hat{q}_{OLS})$ are $n \times 1$ column vectors respectively corresponding to the residual estimator and estimate.

With the assumption that $\varepsilon_i \sim \mathcal{N}(0, \sigma_0^2)$, the nonlinear OLS estimator is consistent—that is, for a sufficiently large sample size, $\mathbb{E}(q_{OLS}) = q_0$—as well as asymptotically normal [36, 37, 38]. In particular, given a large number of data points, the sampling distribution can be accurately approximated as $q_{OLS} \sim \mathcal{N}(q_0, \hat{V}_{OLS})$, where $\hat{V}_{OLS}$ is the OLS estimate of covariance given by

$$\hat{V}_{OLS} = \hat{\sigma}_{OLS}^2 [\chi^T(\hat{q}_{OLS}) \chi(\hat{q}_{OLS})]^{-1}.$$

Here σ_{OLS}^2 is the OLS estimate of error variance and $\chi(q)$ represents the sensitivity matrix

$$\chi_{ik}(\hat{q}_{OLS}) = \frac{\partial f_i(\hat{q}_{OLS})}{\partial q_k}$$

evaluated at the OLS estimate $\hat{q}_{OLS}$. Hence, when we use asymptotic theory to construct the sampling distribution, we assume that our number of data points is sufficiently large to enter the asymptotic regime and simply employ a multivariate Gaussian distribution utilizing the OLS estimates $\hat{q}_{OLS}$ and $\hat{V}_{OLS}$ as the mean and covariance, respectively.

2.1.2 Bootstrapping

Whereas asymptotic analysis is computationally efficient, there are many conditions required for its accurate characterization of the sampling distribution beyond a rough approximation. In the previous section, we assumed that we have a “large enough" sample size to apply asymptotic properties. The number of data points that constitute a sufficiently large sample is often ambiguous, but it is clear that asymptotic theory will not perform well for small data sets. Moreover, many of the results described previously for the nonlinear OLS estimator and its asymptotic properties were derived using a linear Taylor series expansion along with the assumption of local linearity to exploit existing theory for the linear OLS estimator [37, 36]. For highly nonlinear problems, the assumption of local linearity is no longer valid, and the results for the previous section cannot be applied.

Bootstrapping provides an alternative method to construct sampling distributions associated with frequentist estimators. Bootstrapping outperforms asymptotic theory for small sample sizes, and we can relax the assumption that the errors are iid and normally distributed to requiring that they are simply iid. Moreover, bootstrapping techniques are not rooted in linear theory, so we no longer require the assumption of local linearity and may freely employ such methods for highly nonlinear problems [7].

When we apply bootstrapping from a frequentist perspective, we seek to determine the uncertainty of an estimator by constructing its sampling distribution. We again consider the nonlinear OLS estimator, and we estimate the parameters based on (2.3) using n data points. Ideally, we would get a sense of the distribution associated with the OLS parameter estimates by repeatedly resampling to obtain new data points and applying (2.3) to recalculate the parameter estimates. In particular, we would obtain Monte Carlo approximations to the parameter distributions if the number of resampling iterations was large. However, resampling to obtain new data points is often either impossible or impractical. The idea of bootstrapping is to use the n data points in place of the larger population and sample from the original data points with replacement. As detailed in [7], bootstrapping follows these general steps:

1. Sample with replacement from the original n data points; this newly sampled set is called the “bootstrap sample."

2. Compute estimate(s) using the desired estimator with the bootstrap sample.

3. Repeat steps 1 and 2 M times.

For large enough M, we are essentially constructing a Monte Carlo approximation of the estimator sampling distribution, but we are drawing from an empirical distribution, which assigns a probability of $1/n$ to each of the originally-collected data points, in place of the unknown population distribution.

With the idea of bootstrapping in place, it may seem reasonable to approach the problem of constructing parameter pdf's for the Dittus-Boelter model (2.2) by obtaining a large number of bootstrap samples from the collected ordered triples $[Re_i, Pr_i, Nu_i]$, estimating the the parameters q_1, q_2, and q_3 for each bootstrap sample, and using the set of the estimates to obtain a Monte Carlo approximation to the parameter distributions. However, this approach can be theoretically problematic; it treats the design conditions—in this case, Re and Pr—as random rather than fixed, which is inappropriate for experiments necessitating measurements under specific designs [12]. To treat the design conditions as fixed, we utilize the bootstrapping method described in Algorithm 1. This method treats the originally-collected data as fixed and instead resamples the residuals of the fitted model (2.2). The bootstrap samples in Step 3 provide a Monte Carlo approximation to the parameter pdf's. For the Dittus-Boelter problem, we employed $M = 10^5$ bootstrap samples.

Algorithm 1 Construction of Parameter pdf's via Bootstrapping [16]

1. Set $\hat{q}_{OLS}$ equal to the ordinary least square estimate of the parameter vector.
2. For $m = 0, 1, \ldots, M-1$, where M is the number of bootstrapping samples,

    For $j = 0, 1, \ldots, n-1$, where n is the number of data points,

    (a) Construct set of standardized residuals $\{r_j\}_{j=0}^{n}$ as

    $$r_j = \sqrt{\frac{n}{n-p}}\left[y_j - f(x, \hat{q}_{OLS})\right],$$

    where p is the number of parameters, f is the model function, and x is the vector of design conditions.

    (b) Sample from $\{r_j\}_{j=0}^{n}$ with replacement to generate a bootstrap sample of n standardized residuals

    $$\left\{\tilde{r}_0^m, \tilde{r}_1^m, \ldots, \tilde{r}_{n-1}^m\right\}.$$

    (c) Generate synthetic data $y_j^m = f(x, \hat{q}_{OLS}) + \tilde{r}_j^m$.

    (d) Using the synthetic data, calculate the ordinary least squares estimate to obtain $\tilde{q}_m$.

3. This generates bootstrap samples $\left\{\tilde{q}_0, \tilde{q}_1, \ldots, \tilde{q}_{M-1}\right\}$.

2.1.3 Delayed Rejection Adaptive Metropolis (DRAM)

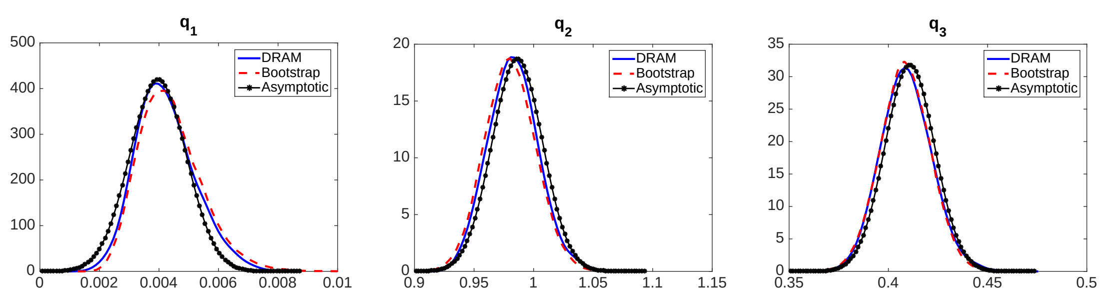
Recall from Chapter 1 that Bayesian parameter estimation entails constructing posterior densities, which represent the "state of knowledge" about the parameter. Since the constructed posterior densities inherently characterize parameter uncertainty, this approach to parameter estimation is natural for the goals of UQ. Here, we employ the DRAM algorithm for Bayesian parameter estimation and, hence, for building parameter pdf's.

The DRAM algorithm [15, 37] is a modified version of the Metropolis-Hastings algorithm, a Markov Chain Monte Carlo (MCMC) technique used to randomly sample from probability distributions. In the case of Bayesian inference, the Metropolis-Hastings algorithm is used to sample from the posterior parameter densities. The DRAM algorithm—detailed in Algorithms 2 and 3 for problems employing a normal likelihood function—adds two additional steps, which correspond to adaptation and delayed rejection. The adaptive step allows for the geometry of the proposal function to be updated as new information about the posterior densities is acquired, and the delayed rejection step improves the mixing of the chains. In order to achieve good exploration of the parameter space via the proposal distribution $\mathcal{N}(\theta^{k-1}, V_{k-1})$, it is important that the covariance matrix reflect the geometry of the parameter space. The incorporation of the adaptation step allows for a poor initial estimate of the covariance to be corrected, enabling a more efficient exploration of the chains.

To implement DRAM, we used OLS estimates as the starting values of q_1, q_2, and q_3. We bounded each of the parameters with a lower limit of zero and an upper limit of two times the nominal values from (2.1); this is standard practice in the nuclear engineering literature when computing uncertainties. We used the least squares estimate of variance $\hat{\sigma}^2_{OLS} = 186.0443$ for the initial value of the error variance. For design parameters, we chose $n_s = 1$, $\sigma_s^2 = \hat{\sigma}^2_{OLS}$, $s_p = 2.38^2/p$, and $k_0 = 100$. With this setup, we implemented the DRAM code from the MATLAB MCMC Toolbox available for download at `http://helios.fmi.fi/~lainema/mcmc/`. After a burn-in period of 10^4 iterations, we reran the code for 10^5 iterations starting with the results from the previous run. We obtained the DRAM estimate error variance by taking the mean of the resulting error variance chain.

2.2 Results

Figure 2.1 shows the parameter pdf's obtained using asymptotic analysis, bootstrapping, and DRAM with 56 points pooled from the four data sets. The resulting distributions are similar for all three methods, but those generated via bootstrapping and DRAM are nearly visually indistinguishable. Since we are using a flat prior and a normal likelihood function with DRAM, the Bayesian technique should only agree with a frequentist method if the problem is linear or if the parameter distributions are Gaussian. However, the Dittus-Boelter model (2.2) is clearly nonlinear, and Figure 2.2 also shows



(a) (b) (c)

Figure 2.1: Distributions for parameters (a) q_1, (b) q_2, and (c) q_3 constructed using asymptotic theory, bootstrapping, and DRAM with all four data sets.

that the q_1 parameter distribution is slightly skewed and, therefore, non-Gaussian. This suggests that the model might exhibit linear behavior in the observed region.

To examine this possibility, we consider—as an alternative to (2.2)—the statistical model

$$Nu(q,Re,Pr) = f(q_{nom},Re,Pr) + Df(q_{nom},Re,Pr)(q-q_{nom}) + \varepsilon \tag{2.5}$$

based on linearization about the nominal parameter values $q_{nom} = [0.023, 0.8, 0.4]$, where

$$f(q,Re,Pr) = q_1 Re^{q_2} Pr^{q_3}$$

and

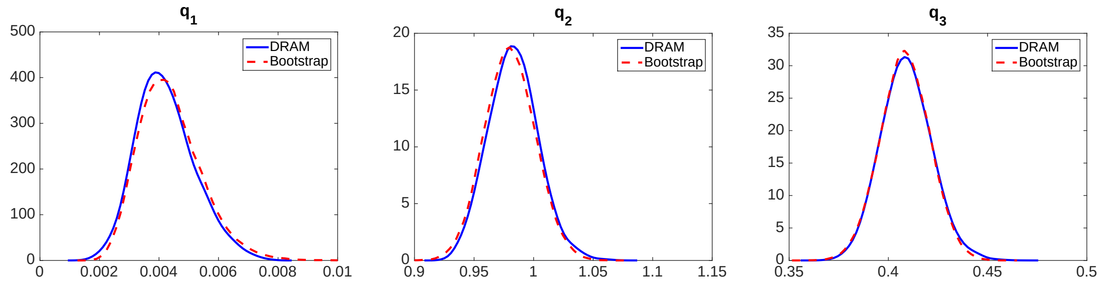
$$Df(q,Re,Pr) = \left[\left.\frac{\partial f}{\partial q_1}\right|_{q,Re,Pr}, \left.\frac{\partial f}{\partial q_2}\right|_{q,Re,Pr}, \left.\frac{\partial f}{\partial q_3}\right|_{q,Re,Pr} \right].$$

Implementing the DRAM algorithm for this model, we used the least squares parameter estimates $\hat{q}_{OLS} = [-0.4457, 2.3509, 1.6019]$ as the starting values of the parameters, and we used the least squares estimate of variance for the initial value of the error variance. No bounds were placed on the parameters. After a burn-in period of 10^4 iterations, we reran the DRAM code for 10^5 iterations starting with the results from the previous run. Using the mean values of the estimated parameter distributions, we calculated the residuals for the nonlinear model (2.2) and the linear model (2.5). A plot comparing the residuals for both models is given in Figure 2.3. The residuals are very similar in pattern and size, suggesting that the two models provide similar fits. Hence, it is likely that the model behavior is approximately linear in the observed region, which explains the agreement of the parameter distributions obtained from bootstrapping and DRAM.

Table 2.1 gives a comparison of the mean values of the three constructed distributions and the nominal parameter values. While the means of all three distributions agree, these values are notably different for parameters q_1 and q_2. Since we also see a reduction in our estimation of uncertainty, this suggests that the parameter estimates for all three methods are an improvement of the nominal values used in (2.1). This reduction in uncertainty is characterized by the 95% confidence intervals for the frequentist methods and the 95% credible intervals for DRAM given in Table 2.2. Note that the intervals derived from the constructed pdf's are much narrower than the nominal interval of uncertainty.

2.3 Model Improvements

Whereas we were able to successfully construct pdf's for the parameters of (2.2) with the results suggesting estimates that are reasonably close to the nominal values, closer examination reveals the limitations of the model. In particular, the pairwise plots generated by DRAM indicate identifiability



(a) (b) (c)

Figure 2.2: Distributions for parameters (a) q_1, (b) q_2, and (c) q_3 constructed using bootstrapping and DRAM with all four data sets.

Table 2.1: Comparison of the nominal values to the means of the parameter pdf's constructed via asymptotic analysis, bootstrapping, and DRAM.

	Asymptotic	Bootstrapping	DRAM	Nominal
q_1	0.0040	0.0044	0.0042	0.023
q_2	0.9863	0.9803	0.9831	0.8
q_3	0.4108	0.4085	0.40913	0.4

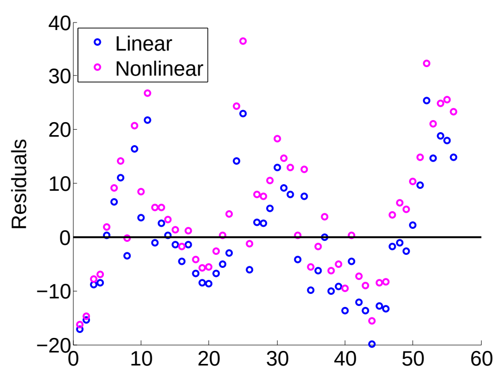


Figure 2.3: Residuals for non-linear model (2.2) and linear model (2.5).

problems. Figure 2.4 shows that q_1 and q_2 have a nearly one-to-one relationship. Hence, if we fix one of the parameters to an arbitrary value, we could find a corresponding value for the other parameter that optimally fits the model to the data. This means that the parameter set $q = [q_1, q_2, q_3]$ is not jointly identifiable in the sense that the parameters cannot be uniquely determined from the data.

Table 2.2: The 95 % confidence intervals for asymptotic theory and bootstrapping as well as the 95% credible intervals for DRAM compared to the nominal uncertainty.

	Asymptotic Intervals	Bootstrapping Intervals	DRAM Intervals	Nominal Uncertainty
q_1	$[0.0021, 0.0059]$	$[0.0023, 0.0065]$	$[0.0025, 0.0064]$	$[0, 0.046]$
q_2	$[0.9436, 1.0291]$	$[0.9377, 1.0228]$	$[0.94401.0262]$	$[0, 1.6]$
q_3	$[0.3857, 0.4359]$	$[0.3836, 0.4335]$	$[0.3853, 0.4353]$	$[0, 0.8]$

We can resolve this issue by fixing either q_1 or q_2 in the model (2.2) and estimating the remaining two parameters.

In addition to identifiability problems, we have the issue of incompatible parameters. To understand incompatible parameters, we present in Figure 2.5 a comparison of two scenarios in which data is modeled by a parameterized version of the McAdams relation [46]

$$F = \xi_1 Re^{\xi_2} + \varepsilon \tag{2.6}$$

where F is the friction factor, Re is the Reynolds number, ξ_1 and ξ_2 are parameters, and ε represents measurement error. In one scenario, the data points from 5 experiments are aligned in such a way that a single model curve—and, hence, a single set of optimal parameters—provides a reasonable fit for all data sets. However, this is not true of the second scenario. While the shape of the curves are similar for all of the experimental data sets, suggesting the same underlying physics, there is not one set of parameter values that will optimally fit all of the data sets. As shown in Figure 2.6, a similar situation arises with the data from [27] for the Dittus-Boelter problem. To remedy this type of situation, we need to account for parameter variability among the individual data sets and allow for individual fits. Ideally, we would also like to quantify “nominal" parameter values that adequately represent all data sets. In Chapter 3, we introduce mixed-effect models as a way to account for variability within a population of parameters. We also present techniques for estimating parameters of mixed-effects models and employ them for an improved version of (2.2), which remedies the nonidentifiability and the incompatibility of the parameters.

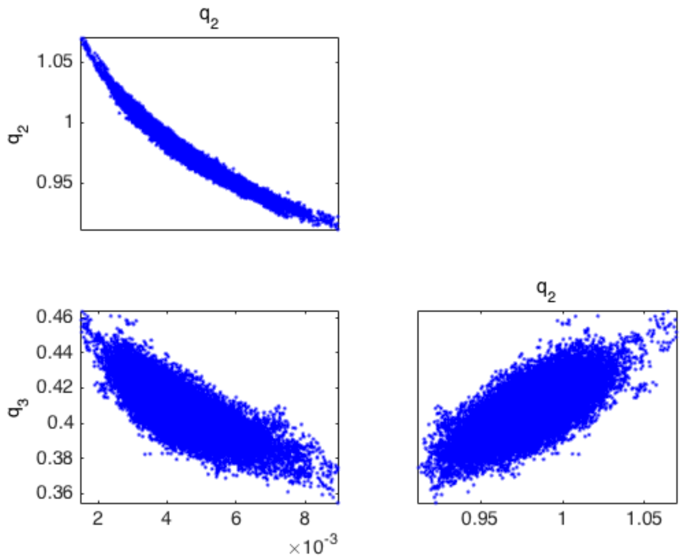


Figure 2.4: Pairwise plots for parameters of (2.2) obtained using DRAM.

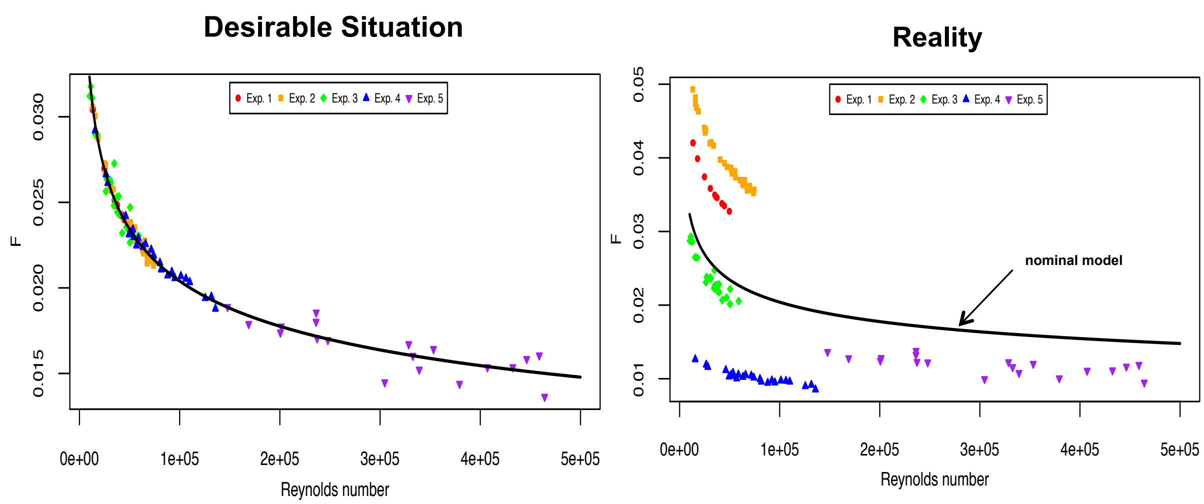


Figure 2.5: Two scenarios involving data collected from multiple experiments modeled by (2.6).

Algorithm 2 Delayed Rejection Adaptive Metropolis with a Normal Likelihood Function and a Flat Prior Distribution [15, 37]

1. Set design parameters n_s, σ_s^2, s_p and k_0 and the number of chain iterates M. Here n_s and σ_s^2 are employed to update the error variance. The design parameter s_p depends on the dimension of the problem; we use $s_p = 2.38^2/p$. Also, k_0 denotes the number of iterations between updates of the covariance matrix.

2. Determine q^0 =arg $\min_q \sum_{i=1}^n [\upsilon_i - f_i(q)]^2$.

3. Set $SS_{q^0} = \sum_{i=1}^n [\nu_i - f_i(q^0)]^2$.

4. Compute initial variance estimate $s_0^2 = \frac{SS_{q0}}{n-p}$, where n is the number of data points and p is the number of parameters.

5. Construct initial variance estimate $V_0 = s_0^2\left[\chi^T(q^0)\chi(q^0)\right]^{-1}$ and set $R_0 = \text{chol}(V_0)$, where the sensitivity matrix has components $\chi_{ij} = \dfrac{\partial f_i(q^0)}{\partial q_j}$.

6. For $k = 1, ..., M$

   (a) Sample $z_k \sim \mathcal{N}(0, I)$.

   (b) Construct candidate $q^* = q^{k-1} + R_{k-1}^T z_k$.
   Note that this is equivalent to sampling $q^* \sim \mathcal{N}(q^{k-1}, V_{k-1})$.

   (c) Sample $u_\alpha \sim \mathcal{U}(0,1)$.

   (d) Compute $SS_{q^*} = \sum_{i=1}^n [\nu_i - f_i(q^*)]^2$.

   (e) Compute

   $$\alpha(q^*|q^{k-1}) = \min\left(1, e^{-\left[SS_{q^*} - SS_{q^{k-1}}\right]/2s_{k-1}^2}\right).$$

   (f) If $u_\alpha < \alpha$,

      Set $q^k = q^*$, $SS_{q^k} = SS_{q^*}$.

   else

      Enter Delayed Rejection Algorithm 3.

   endif

   (g) Update $s_k^2 \sim$ Inv-gamma(a_{val}, b_{val}), where

      $a_{val} = 0.5(n_s + n)$, $b_{val} = 0.5(n_s\sigma_s^2 + SS_{q^k})$.

   (h) If mod $(k, k_0) = 1$,

      Update $V_k = s_p \text{cov}(q^0, q^1, ..., q^k)$ and $R_k = \text{chol}(V_k)$.

   else

      $V_k = V_{k-1}$, $R_k = R_{k-1}$.

   endif

Algorithm 3 Delayed Rejection Component of DRAM with a Normal Likelihood Function [15, 37]

1. Set the design parameter $\gamma_2 < 1$. We set $\gamma_2 = \frac{1}{5}$.

2. Sample $z_k \sim \mathcal{N}(0, I)$.

3. Construct second-stage candidate $q^{*2} = q^{k-1} + \gamma_2 R_{k-1}^T z_k$.
Note that this is equivalent to sampling $q^{*2} \sim \mathcal{N}(q^{k-1}, \gamma_2^2 V_{k-1})$.

4. Sample $u_{\alpha_2} \sim \mathcal{U}(0,1)$.

5. Compute $SS_{q^{*2}} = \sum_{i=1}^n [\nu_i - f_i(q^{*2})]^2$.

6. Compute

$$\alpha_2(q^{*2}|q^{k-1}, q^*) = \min\left(1, \tfrac{\pi(q^{*2}|\nu) J(q^*|q^{*2})[1-\alpha(q^*|q^{*2})]}{\pi(q^{k-1}|\nu) J(q^*|q^{k-1})[1-\alpha(q^*|q^{k-1})]}\right),$$

where π is the normal likelihood function and J is the proposal, or jumping, distribution in (6a-b) of Algorithm 2. Specifically,

$$J(q^a|q^b) = \frac{1}{\sqrt{(2\pi)^p |V|}} \exp\left(-\frac{1}{2}\left[(q^a - q^b) V^{-1} (q^a - q^b)^T\right]\right).$$

7. If $u_{\alpha_2} < \alpha_2$,

    Set $q^k = q^{*2}$, $SS_{q^k} = SS_{q^{*2}}$.

   else

    Set $q^k = q^{k-1}$, $SS_{q^k} = SS_{q^{k-1}}$.

   endif

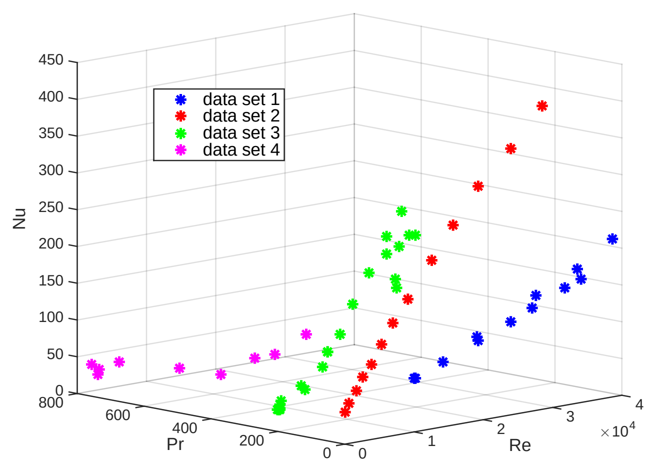


Figure 2.6: Three-dimensional plot of data from [27].

CHAPTER

3

PARAMETER ESTIMATION TECHNIQUES FOR MIXED-EFFECTS MODELS

Varying responses within a population can be described by a mixed-effects model. Such a model includes fixed, population-wide effects as well as random effects, which incorporate individual variation. The statistical mixed-effects model takes the form of

$$y_{ij} = f(x_{ij}; \beta, b_i) + \varepsilon_{ij} \tag{3.1}$$

where, for each individual i, y_{ij} is the jth observation, x_{ij} is the jth vector of independent variables, β is the vector of fixed-effect parameters, b_i denotes the vector of random effects, and ε_{ij} is the measurement error. It is assumed that

$$\begin{aligned} b_i &\sim \mathcal{N}(0, \Psi) \qquad (3.2)\\ \varepsilon_{ij} &\sim \mathcal{N}(0, \sigma^2), \end{aligned}$$

where Ψ is the covariance matrix of the random effects and σ^2 is the variance of the measurement errors. Note that the covariance matrix Ψ is diagonal if the random effects for each individual are assumed to be independent. The quantities to be estimated are the fixed effects β, the random effects b_i, the error variance σ^2, and the elements of the covariance matrix Ψ. Frequently, the effective parameters $\beta_i = \beta + b_i$ for each individual i are estimated in place of the random effects.

3.1 Current Parameter Estimation Techniques

3.1.1 Frequentist Methods

Parameter estimation in the frequentist framework involves constructing an estimator, which assigns optimal parameter values based on the observed data. Recall from Chapter 1 that the parameters in frequentist inference are considered fixed but unknown and that the data are considered realizations of random variables. In Chapter 2, we utilized the Ordinary Least Squares (OLS) estimator to obtain optimal parameter values. Here we employ maximum likelihood estimation.

To define a maximum likelihood estimator, we first must define a likelihood function. Let $f_\Upsilon(\nu; q)$ be a joint probability density function where $q \in \mathbb{Q}$ is an unknown vector of parameters in the admissible parameter space $\mathbb{Q}$, $\Upsilon = [\Upsilon_1, \ldots, \Upsilon_n]$ is the associated random vector, and $\nu = [\nu_1, \ldots, \nu_n]$ are realizations of Υ. Then, the likelihood function $L : \mathbb{Q} \to [0, \infty)$, as defined by [37], is given by

$$L(q) = L(q|\nu) = f_\Upsilon(\nu; q).$$

We note that the likelihood is a function of the parameters as compared to the sampling distribution, which is a function of the data. Using this likelihood function with the assumption that the samples ν_i are iid, we obtain maximum likelihood estimate

$$\hat{q}_{MLE} = \underset{q \in \mathbb{Q}}{\operatorname{argmax}} \prod_{i=1}^{n} f_\Upsilon(\nu_i; q)$$

as in [37].

Let a general linear mixed-effects model with p_F fixed effects and p_R random effects be given by

$$y = X\beta + Zb + \varepsilon \tag{3.3}$$

where y is the $n \times 1$ response vector, X is the $n \times p_F$ design matrix for the fixed effects, β is the $p_F \times 1$ vector of fixed effects, Z is the $n \times p_R$ design matrix for the random effects, b is the $p_R \times 1$ vector of random effects, and ε is the vector of measurement errors. We assume that

$$\begin{aligned} b &\sim \mathcal{N}(0, \Psi) = \mathcal{N}(0, \sigma^2 D(\theta)), \\ \varepsilon &\sim \mathcal{N}(0, \sigma^2 I_n) \end{aligned}$$

where D is a symmetric, positive semidefinite matrix that is parameterized by the vector θ [21].

From a frequentist perspective, the right-hand side of (3.3) contains random variable b as well as fixed but unknown parameters β, σ^2, and θ. We have realizations y for the corresponding vector of random variables Y in the form of data, but the random effects denoted by b are unobserved. Therefore, we must eliminate the dependence on b to obtain a true frequentist likelihood function $L(\beta, b, \sigma^2, \theta|y)$, which depends only on fixed but unknown parameters and observed random variables.

We now derive the likelihood function as detailed in [21, 30]. Since the random effects b_i for each group $i = 1, \ldots, M$ are independent, it follows that

$$L(\beta, \sigma^2, \theta|y) = \prod_{i=1}^{M} p(y_i|\beta, \sigma^2, \theta) = \prod_{i=1}^{M} \int p(y_i|\beta, b_i, \sigma^2, \theta) p(b_i|\sigma^2, \theta)|d b_i. \tag{3.4}$$

Note that the probability density function $p(y|\beta, b, \sigma^2, \theta)$ is a multivariate normal distribution. In particular, we note that

$$y|\beta, b, \sigma^2, \theta \sim \mathcal{N}(X\beta + Zb, \sigma^2 I_n).$$

Thus, we have

$$p(y|\beta, b, \sigma^2, \theta) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(\frac{-\sum_{i=1}^{M} \|y_i - X_i\beta - Zb_i\|^2}{2\sigma^2} \right) = \frac{1}{(2\pi\sigma^2)^{n/2}} \exp\left(\frac{-\|y - X\beta - Zb\|^2}{2\sigma^2} \right).$$

Recall that $b_i \sim N(0, \Psi)$, so

$$p(b|\theta, \sigma^2) = \frac{1}{(2\pi)^{p_R/2}|\Psi|^{1/2}} \exp\left(-\frac{1}{2} b^T \Psi^{-1} b \right) = \frac{1}{(2\pi\sigma^2)^{p_R/2}|D(\theta)|^{1/2}} \exp\left(-\frac{1}{2\sigma^2} b^T (D(\theta))^{-1} b \right).$$

Thus, we have

$$L(\beta, \sigma^2, \theta|y) = \frac{|\det[\Delta(\theta)]|}{(2\pi\sigma^2)^{n/2}} \int \frac{\exp\left[-\left(\|y - X\beta - Zb\|^2 + \|\Delta(\theta)b\|^2 \right) / 2\sigma^2 \right]}{(2\pi\sigma^2)^{p_R/2}} db \tag{3.5}$$

where $\Delta(\theta)$ is any matrix such that $D(\theta)^{-1} = \Delta(\theta)^T \Delta(\theta)$. One possibility for $\Delta(\theta)$ is to use the Cholesky factorization of $D(\theta)^{-1}$. Note that evaluating the integral in (3.5) will still leave occurrences of b in the right hand side. To eliminate the dependence on b, we find the conditional modes of the random effects given the data. We first let

$$r^2(\beta, b, \theta) = b^T \Delta(\theta)^T \Delta(\theta) b + (y - X\beta - Zb)^T (y - X\beta - Zb).$$

Then, the vector of conditional modes b^* is the vector for which the values of the random effects vector b satisfy

$$\left.\frac{\partial r^2(\beta, b, \theta)}{\partial b}\right|_{b^*} = 0$$

for given β and θ. Now, by evaluating the integral in (3.5) and substituting in the vector of conditional modes b^*, we obtain the likelihood function

$$L(\beta, \sigma^2, \theta | y) = \frac{|\det[\Delta(\theta)]|}{(2\pi\sigma^2)^{n/2}} \exp\left\{-\frac{1}{2\sigma^2} r^2(\beta, b^*(\beta, \theta), \theta)\right\} \frac{1}{|\Delta^T \Delta + Z^T Z|^{1/2}}. \tag{3.6}$$

While (3.6) represents a valid likelihood function, a profiled likelihood, which reformulates (3.6) to exclusively be parameterized by θ, is generally used in practice for numerical optimization [21]. This is achieved by deriving formulas for the conditional estimates $\beta^*(\theta)$ and $\sigma^{2*}(\theta)$, which maximize $L(\beta, \sigma^2, \theta | y)$ for a given value of θ. We then substitute in the conditional estimates to obtain the profiled likelihood $L(\theta | y) = L\left(\beta^*(\theta), \sigma^{2*}(\theta), \theta | y\right)$ [30].

MATLAB has two functions for computing the maximum likelihood estimates for linear mixed-effects problems, namely `fitlme` and `fitlmematrix` [22, 23]. The `fitlme` function uses an input array of data to fit the parameters of a user-provided formula. The `fitlmematrix` function takes input arguments in the form of matrices as defined by X, Z, and y in (3.3) along with an $n \times 1$ grouping vector. Use of the profiled likelihood for MLE is the default for both functions with each function having the option to instead employ restricted maximum likelihood estimation (REML). For mixed-effects models, maximum likelihood estimates of the variance parameters tend to underestimate the true values, especially when these values are small [21, 30]. REML estimation has the benefit of being unbiased for the variance parameters; however, we cannot use likelihood ratio tests such as information criteria to compare mixed-effects models using a restricted likelihood function. Since use of information criteria is important to the mixed-effects parameter subset selection algorithm introduced in Chapter 4, we exclusively use the full profiled likelihood instead of the restricted likelihood throughout this dissertation. For optimization, `fitlmematrix` and `fitlme` employ a quasi-Newton optimizer as the default setting, but both functions have the option of using `fminunc` if the Optimzation Toolbox is installed.

For nonlinear mixed-effect models—that is, models of the form (3.1) when f is nonlinear—the integral in (3.4) generally does not have a closed form, so the MLE estimates cannot be obtained directly. For these cases, the MATLAB Statistics Toolbox has two options for obtaining MLE estimates: `nlmefit` and `nlmefitsa` [24, 25]. The `nlmefit` function uses an approximation to the likelihood function paired with an optimizer to obtain parameter estimates. There are four options for the likelihood approximation: (i) the linear mixed-effects model likelihood at the current conditional estimates of the fixed and random effects, (ii) the linear mixed-effects model restricted likelihood at the current conditional estimates of the fixed and random effects, (iii) the first-order Laplacian approximation without random effects, and (iv) the first-order Laplacian approximation with conditional estimates of the random effects. The default choice is the linear mixed-effects likelihood. MATLAB's `fminsearch` is used to optimize the likelihood function. If the Optimization Toolbox is installed, `fminunc` may alternatively be used as the optimizer.

Instead of employing an approximate likelihood function, `nlmefitsa` uses a stochastic approximation expectation-maximization (SAEM) algorithm to find the parameter estimates. Use of the standard expectation-maximization (EM) algorithm is common for problems with incomplete data. In the case of mixed-effects models, the unobserved random effects are considered to be missing or incomplete data. The EM algorithm is an iterative process. In the expectation step, we formulate the expected value of the log-likelihood function based on the conditional distribution of the random effects given the observed data and the current parameter estimates. In the maximization step, we recalibrate the parameters by maximizing the conditional expectation from the previous step. More details on the EM algorithm are provided in [20]. Whereas the EM algorithm is a useful tool, the standard version of the algorithm is problematic for nonlinear mixed-effects models, which do not have a closed form for the likelihood; the expectation step of the algorithm cannot be done without a full likelihood function [11]. The SAEM algorithm remedies this dilemma by stochastically approximating the expectation for the E step of the EM algorithm [25].

3.1.2 Bayesian Parameter Estimation: Gibbs Sampling

Bayesian parameter estimation for mixed-effects modeling typically utilizes Gibbs sampling, a Markov Chain Monte Carlo (MCMC) method for obtaining random samples from a joint probability density function that is either unknown or difficult to sample. In particular, Gibbs sampling relies on drawing from the conditional distributions of each of the variables. We utilize this technique to estimate the mixed-effects parameters defined by (3.1) and (3.2). For the purpose of Bayesian inference, we assume that the parameters have the prior distributions

$$\beta \sim \mathcal{N}(\beta_0, \Sigma_0)\,,\ \sigma^{-2} \sim \text{Gamma}(\nu_0, \tau_0)\ ,\ b_i \sim \mathcal{N}(0, \Psi)\,,\ \Psi \sim \text{Inv-Wishart}(\Psi_0, \rho_0).$$

In the case of linear mixed-effects problems, all of the conditional distributions required for Gibbs sampling can be completely derived, but with nonlinear mixed-effects problems, the conditional distribution for $\beta_i = \beta + b_i$ has no closed form. Hence, we must do a Metropolis-within-Gibbs step to estimate the effective parameters β_i for each individual i. As derived in [49], the full conditional distributions for the nonlinear mixed-effects model parameters are

$$[\beta|\sigma^2,\Psi,\beta_*,y] \sim \mathcal{N}\left((n_g\Psi^{-1}+\Sigma_0^{-1})^{-1}\left(\Psi^{-1}\sum_{i=1}^{n_g}\beta_i+\Sigma_0^{-1}\beta_0\right),(n_g\Psi^{-1}+\Sigma_0^{-1})^{-1}\right),$$

$$[\sigma^{-2}|\beta,\Psi,\beta_*,y] \sim \text{Gamma}\left(\nu_0+\frac{1}{2}\sum_{i=1}^{n_g}n_i,\left[\tau_0+\frac{1}{2}\sum_{i=1}^{n_g}\sum_{j=1}^{n_i}(y_{ij}-f_{ij}(x_{ij};\beta_i))^2\right]^{-1}\right),$$

$$[\Psi|\sigma,\beta,\beta_*,y] \sim \text{Inv-Wishart}\big((\beta_i-\beta)(\beta_i-\beta)^T+\Psi_0,n_g+\rho_0\big),$$

with

$$f(\beta_i|\beta_*,\sigma,D,y) \propto \exp\left(-\frac{\sigma^{-2}}{2}\sum_{j=1}^{n_i}(y_{ij}-f_{ij}(x_{ij};\beta_i))^2-\frac{1}{2}(\beta_i-\beta)^T\Psi^{-1}(\beta_i-\beta)\right).$$

3.1.3 Bayesian Parameter Estimation: DRAM

Whereas Gibbs sampling is a commonly-applied approach to Bayesian inference for mixed-effects models, it suffers from its chain samples being correlated to neighboring samples. Some efforts have been made to remedy this, including thinning the chains; however, the Delayed Rejection Adaptive Metropolis (DRAM) algorithm will typically outperform Gibbs sampling in the case of highly correlated parameters. Thus, we have augmented the standard DRAM algorithm to construct a version that is appropriate for mixed-effects models. There are current implementations of mixed-effects DRAM algorithm for problems with diagonal random effects covariance matrices. In particular, the MATLAB MCMC Toolbox DRAM code has an option for this that is enacted by `localflag=2`. However, we have made novel changes to generalize the DRAM algorithm to mixed-effects models with non-diagonal Ψ matrices. We modified the Metropolis-within-Gibbs step described in the previous section to be a DRAM-within-Metropolis step, updating all additional parameters via Gibbs sampling. The mixed-effects version of DRAM is given in Algorithms 4 and 5.

Algorithm 4 Delayed Rejection Adaptive Metropolis for Mixed-Effects Models adapted from [15, 37]

1. Set design parameters n_s, σ_s^2, ρ_0, k_0, and the number of chain iterates M.

2. Obtain initial hyperparameter estimates β^0 and Ψ_0 for the vector of parameter means and covariance matrix, respectively. Also, obtain initial estimates $\beta_i^0 = \beta^0 + b_i^0$ for the effective parameters for all $i = 1, 2, \ldots, n$ and s_0^2 for the error variance. Set $q^0 = [\beta_1^0, \ldots, \beta_n^0]$. We compute these estimates via freqentist techniques.

3. Construct the $n_g \cdot p_R \times n_g \cdot p_R$ initial covariance estimate V_0 for the full set of effective parameters where p_R is the number of random effects and n_g is the total number of groups or individuals. Set $R_0 = \text{chol}(V_0)$. To characterize the basic geometry of the problem, we employ $V_0 = \text{diag}[V_1, V_2, \ldots, V_n]$, where $V_i = \text{diag}\left[(0.05 \cdot \beta_{i1}^0)^2, (0.05 \cdot \beta_{i2}^0)^2, \ldots, (0.05 \cdot \beta_{ir_R}^0)^2\right]$ for $i = 1, 2, \ldots, n$.

4. For $k = 1, \ldots, M$

   (a) Sample $z_k \sim \mathcal{N}(0, I)$.

   (b) Construct candidate $q^* = [\beta_1^*, \ldots, \beta_n^*] = q^{k-1} + R^T z_k$ where $R = \text{chol}(V_k)$.
   Note that this is equivalent to sampling $q^* \sim N(q^{k-1}, V_k)$.

   (c) Sample $u_\alpha \sim \mathcal{U}(0, 1)$.

   (d) Compute $\alpha(q^*|q^{k-1}) = \min\left(1, \dfrac{\pi(y|q^*)\pi_0(q^*)}{\pi(y|q^{k-1})\pi_0(q^{k-1})}\right)$ using likelihood function π and prior π_0.

   (e) If $u_\alpha < \alpha$,

      Set $q^k = q^*$.

   else

      Enter Delayed Rejection Algorithm 5.

   endif

   (f) Compute $SS_k = \sum_{i=1}^n \sum_{j=1}^{n_i} (y_{ij} - f_{ij}(\beta_i))^2$.

   (g) Update $s_k^2 \sim \text{Inv-gamma}(a_{val}, b_{val})$, where

$$a_{val} = 0.5\left(n_s + \sum_{i=1}^n n_i\right), \; b_{val} = 0.5(n_s\sigma_s^2 + SS_k).$$

   We set σ_s^2 equal to the frequentist estimate of error variance and choose $n_s = 1$, which is consistent with a non-informative prior [37].

(Continued on the next page)

Algorithm 4 Delayed Rejection Adaptive Metropolis for Mixed-Effects Models (continued)

(h) Update $\beta^k \sim N\left((n\Psi^{-1}+\Sigma_0^{-1})^{-1}\left(\Psi^{-1}\sum_{i=1}^n \beta_i + \Sigma_0^{-1}\beta_0\right), (n\Psi^{-1}+\Sigma_0^{-1})^{-1}\right)$ where the prior on the hyperparameter vector β is $N(\beta_0, \Sigma_0)$.

(i) Update $\Psi_k \sim$ Inv-Wishart$\left(n\Psi_0 + \sum_{i=1}^n (\beta_i^k - \beta^k)(\beta_i^k - \beta^k)^T, n+\rho_0\right)$ where β_i^k is the $p_r \times 1$ column vector of the current effective parameters for the ith data set. To employ a non-informative prior in the sense of utilizing the flattest distribution, ρ_0 is set equal to the number of random effects.

(j) If $\mod(k, k_0) = 1$

Update $V_k = s_p \text{cov}(q^0, q^1, ..., q^k)$ and $R_k = \text{chol}(V_k)$.

else

$V_k = V_{k-1}$.

endif

3.1.4 Nonlinear Example

For the purposes of verification, we compare the results of parameter estimation via Gibbs sampling, `nlmefit` and mixed-effects DRAM. In this example, we use synthetic data to verify the effectiveness of our methods for a nonlinearly parameterized mixed-effects problem. We consider the classic orange tree growth model

$$y_{ij} = \frac{\beta_1 + b_{1i}}{\left(1+e^{-[t_{ij}-(\beta_2+b_{2i})]/(\beta_3+b_{3i})}\right)} + \varepsilon_{ij} \tag{3.7}$$
$$b_i \sim \mathcal{N}(0, \Psi),\ \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I)$$

from [24] where, for the jth data point in the ith data set, y_{ij} is the tree circumference in millimeters, t_{ij} is the time in days, $\beta = [\beta_1, \beta_2, \beta_3]$ are the fixed effects, and $b_i = [b_{1i}, b_{2i}, b_{3i}]$ are the random effects. We generated synthetic data for $n_g = 5$ individuals using the model (3.7) with error variance drawn from $\varepsilon_{ij} \sim \mathcal{N}(0,1)$,

$$t_i = \begin{bmatrix} 118 & 484 & 664 & 1004 & 1231 & 1372 & 1582 \end{bmatrix}^T$$

Algorithm 5 Delayed Rejection Component [15, 37]

1. Set the design parameter $\gamma_2 < 1$. We set $\gamma_2 = \frac{1}{5}$.

2. Sample $z_k \sim \mathcal{N}(0, I)$.

3. Construct second-stage candidate $q^{*2} = q^{k-1} + \gamma_2 R_k^T z_k$ where $R_k = \text{chol}(V_k)$.
   Note that this is equivalent to sampling $q^{*2} \sim N(q^{k-1}, \gamma_2^2 V_k)$.

4. Sample $u_{\alpha_2} \sim \mathcal{U}(0,1)$.

5. Compute

   $$\alpha_2(q^{*2}|q^{k-1}, q^*) = \min\left(1, \tfrac{\pi(q^{*2}|\nu)J(q^*|q^{*2})[1-\alpha(q^*|q^{*2})]}{\pi(q^{k-1}|\nu)J(q^*|q^{k-1})[1-\alpha(q^*|q^{k-1})]}\right),$$

   where J is the proposal, or jumping, distribution defined by (4a-b) in Algorithm 4. Specifically,

   $$J(q^a|q^b) = \frac{1}{\sqrt{(2\pi)^{n_g p_R}|V|}} \exp\left(-\frac{1}{2}\left[(q^a - q^b)V^{-1}(q^a - q^b)^T\right]\right).$$

6. If $u_{\alpha_2} < \alpha_2$,

   Set $q^k = q^{*2}$.

   else

   Set $q^k = q^{k-1}$.

   endif

for all $i = 1, \ldots, 5$, $b_i = [b_{1i}, b_{2i}, b_{3i}]$ drawn from $\mathcal{N}(0, \Psi)$ with

$$\Psi = \begin{bmatrix} 15 & 0 & 0 \\ 0 & 100 & 0 \\ 0 & 0 & 50 \end{bmatrix}, \tag{3.8}$$

and $\beta_1 = 175$, $\beta_2 = 800$, and $\beta_3 = 300$.

To implement Gibbs sampling, we used chains constructed from 10^7 iterations. Using the conditional distributions from Section 3.1.2, we drew each new chain value out of the specified probability density, setting all other parameter values equal to their latest chain sample. The chain starting values for the fixed effects, effective parameters, and error variance were set equal to the frequentist estimates obtained from MATLAB function `nlmefit`. We employed the true value of Ψ, given by (3.8), as the starting value for the random effects covariance matrix. To employ a flat prior for σ^2, we set ν_0 and τ_0 to be small. In particular, we used $\nu_0 = 0.001$ and $\tau_0 = 0.001$. For the random effects covariance matrix, we set $\rho_0 = 100$, which suggests a highly informative prior for Ψ. The most noninformative choice, in the sense that it corresponds to the flattest distribution, is to let ρ_0 be equal to the number of random effects. We initially employed a noninformative prior; however, when we used $\rho_0 = 3$, the estimate we obtained for the random effects covariance matrix was

$$10^3 \times \begin{bmatrix} 0.9376 & 0.9264 & 0.9264 \\ 0.9264 & 1.0015 & 0.9264 \\ 0.9264 & 0.9264 & 0.9639 \end{bmatrix},$$

which is not close to the true value of Ψ used to generate the data. Using $\rho_0 = 100$ produces much better results; this is shown in Table 3.1, which gives a comparison of the Gibbs sampling and `nlmefit` parameters estimates. Note that the output for `nlmefit` gives estimates of b_i instead of the effective parameters, but we have calculated the effective parameters $\beta + b_i$ and used these values in the table for comparison to the Gibbs sampling results. For Gibbs sampling, the reported parameter estimates are the mean values of the chains. Aside from the random effects covariance matrix, the parameter estimates are very similar, suggesting that the methods agree. When we employed the `nlmefit` function, we activated the option of assuming a diagonal random effects covariance matrix, which explains its success in correctly placing zeros in the off-diagonal entries of its Ψ estimate, but the Gibbs sampler with the informative prior more closely approximated the true Ψ matrix given in (3.8). However, even the Gibbs sampling estimate of Ψ is not particularly close.

These results suggest that a flat prior may not be the best choice for Gibbs sampling when starting with a reasonably accurate approximation for the random effects covariance matrix. Further research is required to determine a recommended methodology. Possible techniques could involve utilizing informative priors or employing empirical Bayes methods to initially estimate the random effects covariance matrix and then treat it as a constant during Gibbs sampling.

We also employed the mixed-effects version of the DRAM algorithm given in Algorithms 4 and 5 to estimate the effective parameters for the model (3.7). We used likelihood function

$$\pi(y|q) = \pi(y|[\beta_1, \ldots, \beta_5]) = \exp \sum_{i=1}^{5} \left[-\frac{\sigma^{-2}}{2} \sum_{j=1}^{n_i} \left[y_{ij} - f_{ij}(x_{ij}; \beta_i) \right]^2 \right]$$

and prior function

$$\pi_0(q) = \pi_0([\beta_1, \ldots, \beta_5]) = \exp \sum_{i=1}^{5} \left[-\frac{1}{2} (\beta_i - \beta)^T \Psi^{-1} (\beta_i - \beta) \right].$$

For the chain starting values, we set the effective parameters equal to their frequentist estimates from `nlmefit`, and we bound the effective parameters to be greater than zero. For the hyperpa-

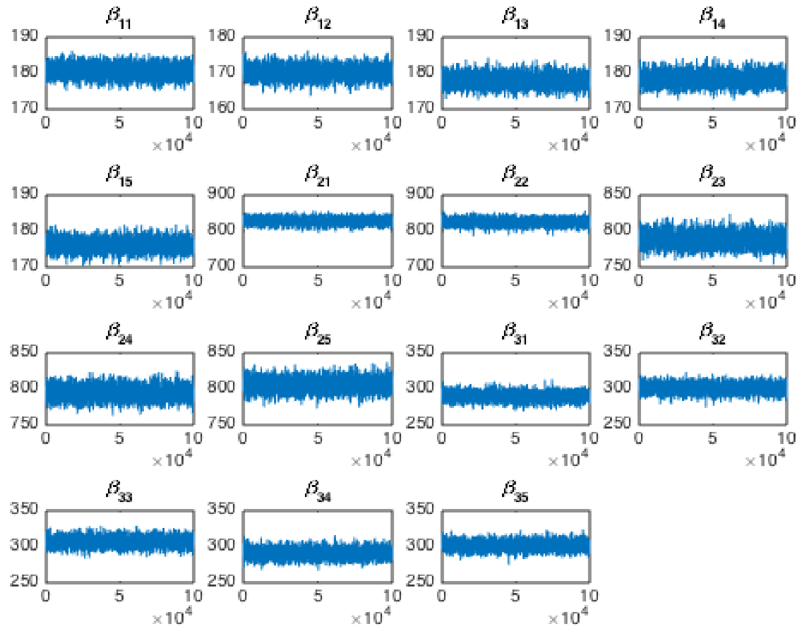


Figure 3.1: Chains generated using DRAM for the effective parameters of the orange tree model (3.7).

rameters, we used the true random effects covariance matrix (3.8) as Ψ_0. We set $\rho_0 = 100$ to employ an informative prior for the covariance matrix as we did with Gibbs sampling. We employed a noninformative uniform hyperprior for β, which is equivalent to setting β_0 equal to the frequentist estimate of the fixed effects and

$$\Sigma_0 = \begin{bmatrix} \infty & & \\ & \ddots & \\ & & \infty \end{bmatrix}.$$

After a burn-in period of 10^6, we constructed final parameter chains of length 10^5. Visual inspection suggests that the chains (Figure 3.1) and the hyperchains (Figure 3.2) are burned in. Moreover, the entries in Table 3.1 indicate that for the nonlinear orange-tree growth model (3.7) the DRAM effective parameter estimates, constructed from the mean chain values, agree with the Gibbs sampling results. The fixed-effect estimates also agree with Gibbs sampling, and the DRAM estimate of Ψ is significantly closer to the true matrix than those obtained using both Gibbs sampling and `nlmefit`.

Table 3.1: Estimated parameter values for (3.7) from `nlmefit`, Gibbs sampling, and DRAM.

	`nlmefit`	Gibbs	DRAM
β_1	176.8110	176.7097	176.7428
β_2	808.7220	808.2492	808.4321
β_3	298.2351	298.0403	298.1843
β_{11}	181.9081	181.1310	180.7033
β_{12}	172.1659	169.5569	170.3078
β_{13}	176.9944	178.0913	178.0310
β_{14}	176.6842	178.2312	178.4585
β_{15}	176.3024	176.5359	176.2340
β_{21}	837.4897	830.0412	828.9037
β_{22}	837.0985	820.9839	825.1117
β_{23}	779.9052	789.6234	788.3719
β_{24}	783.6309	793.2692	793.9753
β_{25}	805.4958	807.3115	805.8295
β_{31}	291.1670	291.8160	289.9685
β_{32}	306.5638	298.6679	301.0232
β_{33}	306.6009	307.2777	307.4212
β_{34}	284.3055	290.1250	290.7398
β_{35}	302.5384	302.3099	301.7683
σ^2	0.7418	1.5737	1.4627
Ψ	$\begin{bmatrix} 11.34 & 0 & 0 \\ 0 & 674.16 & 0 \\ 0 & 0 & 104.83 \end{bmatrix}$	$\begin{bmatrix} 35.44 & 20.59 & 20.59 \\ 20.59 & 119.62 & 20.59 \\ 20.59 & 20.59 & 70.10 \end{bmatrix}$	$\begin{bmatrix} 15.69 & -0.47 & -0.36 \\ -0.47 & 115.14 & -1.50 \\ -0.36 & -1.50 & 53.27 \end{bmatrix}$

3.2 Revised Dittus-Boelter Model

In Chapter 2, we introduced the parameterized version of the Dittus-Boelter equation and noted two problems with the model. In particular, the parameters q_1 and q_2 were not mutually identifiable. We can remedy this by fixing q_1 to be its nominal value in (2.1). Additionally, we had the problem of incompatible parameters; no one set of parameter estimates would simultaneously provide a good fit for all of the data sets. We switch to using a mixed-effects model to permit individual fits as well as population-wide parameter estimates. The resulting model is

$$Nu_{ij} = 0.023 Re^{(\beta_2 + b_{2i})} Pr^{(\beta_3 + b_{3i})} + \varepsilon_{ij}, \tag{3.9}$$

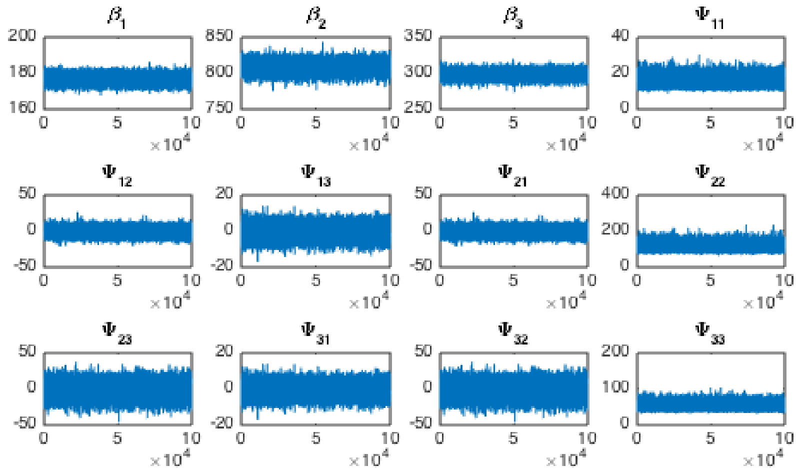


Figure 3.2: Hyperchains generated using DRAM for the components of β and Ψ.

where Nu is the Nusselt number, Pr is the Prandtl number, Re is the Reynolds number, and for the jth observation of the ith data set β_1 and β_2 are the fixed effects, $b_i = [b_{2i}, b_{3i}]$ is the vector of random effects for each data set, and ε_{ij} is the measurement error. We assume that

$$r_i \sim N(0, \Psi),$$
$$\varepsilon_{ij} \sim N(0, \sigma^2).$$

We now estimate the fixed effects, random effects, random effects covariance matrix, and measurement error for the revised Dittus-Boelter model (3.9).

As with the orange tree example, we first perform frequentist parameter estimation to obtain starting values for chains corresponding to the effective parameters $\beta_i = [\beta_{2i}, \beta_{3i}]$. For the purposes of verification, we employed both `nlmefit` and `nlmefitsa`. The resulting parameter estimates are given in Table 3.2. Note that the values are very similar, suggesting that the two methods agree. Moreover, the resulting model fits and residuals are visually indistinguishable. Figure 3.3 shows these plots from `nlmefit`; the identical plots from `nlmefitsa` are omitted. Note that the parameter estimates provide a good fit to the data and that the residuals are fairly iid for the combined data set.

Whereas frequentist parameter estimation is an important first step in the mixed-effects version of the DRAM algorithm, the initial parameter estimates also happened to uncover an insignificant parameter for the Dittus-Boelter model (3.9). Observe that the estimated values of b_{3i} from both `nlmefit` and `nlmefitsa` are essentially zero for all four groups. This implies that this random effect parameter is unnecessary and can be removed from the model, which will lessen the computational cost of the DRAM algorithm. We now have

$$Nu_{ij} = 0.023Re^{(\beta_2 + b_{2i})}Pr^{\beta_3} + \varepsilon_{ij} \tag{3.10}$$

where Nu is the Nusselt number, Pr is the Prandtl number, Re is the Reynolds number, and for the jth observation of the ith data set β_2 and β_3 are the fixed effects, b_{2i} is the single random effects for each data set, and ε_{ij} is the measurement error. We assume that

$$\begin{aligned} b_{2i} &\sim \mathcal{N}(0, \psi^2), \\ \varepsilon_{ij} &\sim \mathcal{N}(0, \sigma^2). \end{aligned}$$

Since we have a single random effect, the covariance matrix is 1×1, so it is simply a variance parameter, which we denote as ψ^2.

Using this new model (3.10), we employ the mixed effects DRAM algorithm for parameter estimation. Note that for the DRAM algorithm, we estimate the effective parameter $\beta_{2i} = \beta_2 + b_{2i}$ instead of the random effect b_{2i}. Also, β_3 no longer has a random effect, so we construct a DRAM chain for β_3 instead of β_{3i}. The updating of β_3 is equivalent to the fixed-effects DRAM process described in Algorithms 2 and 3, and we do not have any corresponding hyperparameters to update for β_3. Also, the covariance matrix V_k will be 5×5 with one diagonal entry corresponding to the variance of the parameter β_3 and the remaining 4 diagonal entries corresponding to the variance of the effective parameters β_{2i} for $i = 1, \ldots, 4$.

For starting values of the chains, we used the `nlmefit` parameter estimates. We employed likelihood function

$$\pi(y|q) = \pi(y|[\beta_{21}, \ldots, \beta_{24}, \beta_3]) = \exp \sum_{i=1}^{4} \left[-\frac{\sigma^{-2}}{2} \sum_{j=1}^{n_i} \big(Nu_{ij} - f_{ij}(Re_{ij}, Pr_{ij}; \beta_{2i}, \beta_3) \big)^2 \right]$$

and prior function

$$\pi_0(q) = \pi_0([\beta_{21}, \ldots, \beta_{24}, \beta_3]) = \exp \sum_{i=1}^{4} \left[-\frac{1}{2\psi^2} (\beta_{2i} - \beta_2)^T (\beta_{2i} - \beta_2) \right] p_0(\beta_3),$$

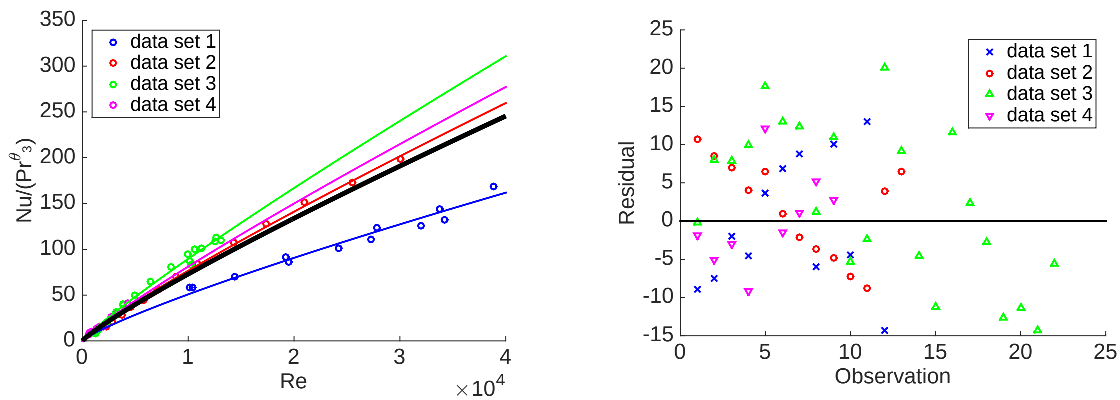
where p_0 is the flat prior on the interval $[0, \infty)$ used for β_3. For the chain starting values, we used the frequentist estimates from `nlmefit`, and we bounded β_{2i} and β_3 to be greater than zero. For the hyperparameters, we used the `nlemfit` estimate of the b_{2i} variance as ψ_0^2, and we set $\rho_0 = 3$ to invoke a fairly noninformative prior. We also employed the noninformative uniform hyperprior described in the previous section for β_2.

After a burn-in period of 10^5, we reran the DRAM code to obtain final chains of length 10^4. The

Table 3.2: Estimated parameter values for (3.9) from `nlmefit` and `nlmefitsa`.

	`nlmefit`	`nlmefitsa`
β_2	0.8758	0.8753
β_3	0.2050	0.2064
β_{21}	−0.0397	−0.0393
β_{22}	0.0053	0.0053
β_{23}	0.0224	0.0223
β_{24}	0.0119	0.0115
β_{31}	−0.0000	−0.0000
β_{32}	−0.0000	−0.0000
β_{33}	0.0000	0.0000
β_{34}	0.0000	−0.0000
σ^2	77.8791	77.8806
Ψ	$\begin{bmatrix} 0.0.00057 & 0 \\ 0 & 0.0000 \end{bmatrix}$	$\begin{bmatrix} 0.00055 & 0 \\ 0 & 0.0000 \end{bmatrix}$

plots in Figures 3.4 and 3.5 indicate that both the chains and hyperchains are burned in. Figure 3.6 shows the model fit and residuals obtained using the mean DRAM chain values. Note that these are visually indistinguishable from those produced by the frequentist methods. Moreover, Table 3.3 indicates that the `nlmefit` and DRAM parameter estimates for model (3.10) agree. While we have successfully estimated the parameters of the revised Dittus-Boelter model (3.10), the fortuitous discovery of the insignificant random effect highlights the need for rigorous parameter selection for mixed-effects models. In Chapter 4, we address the current issues with mixed-effects parameter selection and introduce a new parameter subset selection algorithm.



(a) (b)

Figure 3.3: (a) Model fit and (b) residuals for (3.10) using the parameter estimates from `nlmefit`. The fit and residuals obtained using `nlmefitsa` are not pictured because they are visually indistinguishable from those shown here.

Table 3.3: Estimated parameter values for (3.9) from `nlmefit` and `nlmefitsa`.

	`nlmefit`	DRAM
β_2	0.8758	0.8773
β_3	0.2050	0.2015
β_{21}	−0.0397	−0.0409
β_{22}	0.0053	0.0050
β_{23}	0.0224	0.0226
β_{24}	0.0119	0.0128
σ^2	77.7891	81.0284
ψ^2	0.00057	0.0010

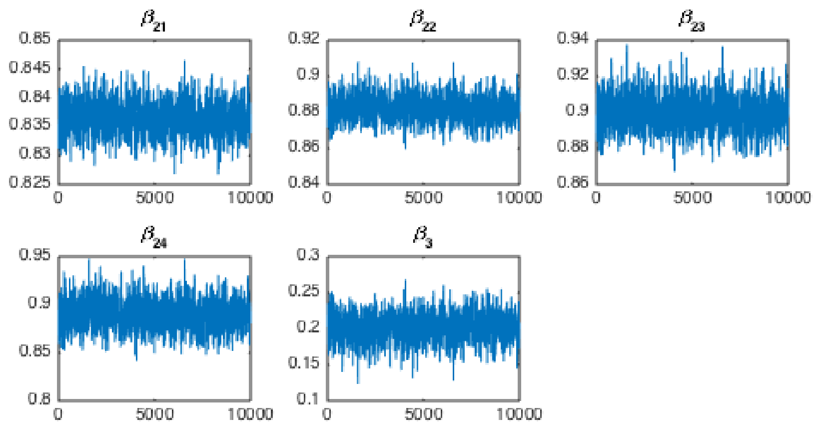


Figure 3.4: Chains generated using DRAM for $\beta_{2i} = \beta_2 + b_{2i}$ and β_3 in model (3.10).

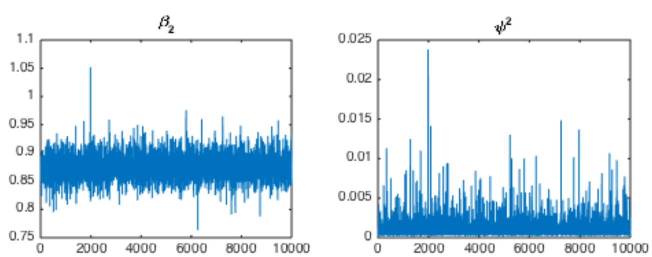
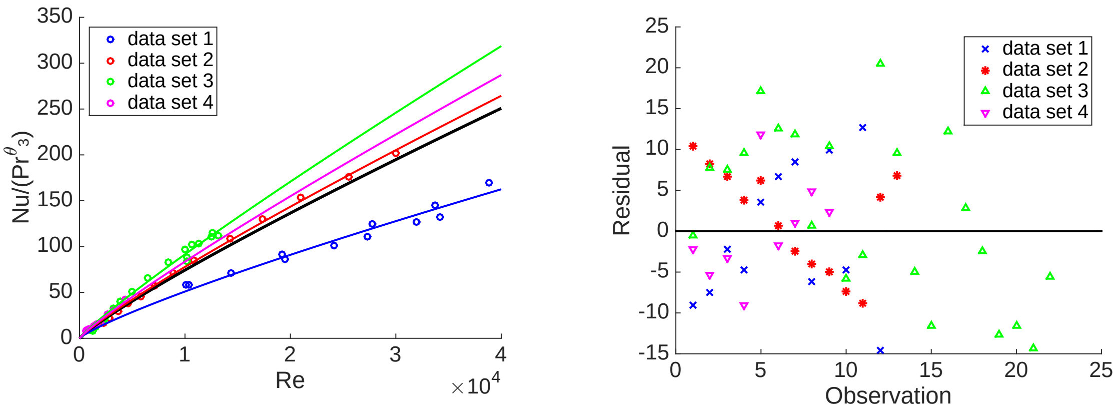


Figure 3.5: Hyperchains generated using DRAM for β_2 and ψ^2 for model (3.10).



(a) (b)

Figure 3.6: (a) The model fit and (b) residuals for (3.10) using the mean DRAM chain values as the parameter estimates.

CHAPTER

4

A PARAMETER SUBSET SELECTION ALGORITHM FOR MIXED-EFFECTS MODELS

4.1 Introduction

As detailed in the previous chapter, mixed-effects models provide one way to accommodate parameter values that vary among individuals in a population. In this chapter, we specifically examine the challenges of parameter selection for mixed-effects models. To illustrate the associated issues, we consider two examples with linearly and nonlinearly parameterized parameters. The contents of this chapter have been submitted for publication [35].

We first consider the problem of modeling the heights of boys measured over time as presented in [50]. Height measurements of 26 boys were recorded at nine occasions over time. The data plotted in Figure 4.1 shows that all of the boys exhibit a linear growth pattern, but no one single choice of an intercept will provide a good fit for all of the individuals since they start at different heights. To quantify the boys' growth in a manner that incorporates the variability in their initial heights, we consider a linearly parameterized statistical model of the form

$$
\begin{aligned}
y_{ij} &= (\beta_0 + b_i) + \beta_1 x_{ij} + \varepsilon_{ij}\ ,\ i = 1, \dots, M\ ,\ j = 1, \dots, n_i \\
&= f(x_{ij}; \beta, b_i) + \varepsilon_{ij}.
\end{aligned}
\tag{4.1}
$$

Here y_{ij} denotes the height of the ith boy at time j, x_{ij} quantifies the time when the measurement was taken, and ε_{ij} is the observation error associated with the measurement. The fixed effect parameters $\beta = [\beta_0, \beta_1]$ apply to the population of all data sets and β_0 can be interpreted as the population average for the intercept, or starting height. The random effect b_i represents the variation in the starting height of the ith boy from the population mean. We typically assume that $b_i \sim N(0, \Psi)$, where the covariance matrix Ψ specifies the variability and correlation in initial height. It is also standard to assume that measurement errors are independent and identically distributed (iid), $\varepsilon_{ij} \sim N(0, \sigma^2)$, and that ε_{ij} and b_i are independent.

Alternatively, we could also choose to include a random effect to account for individual variation of the slope, or the growth rate. In this case, the model would be

$$
y_{ij} = (\beta_0 + b_{0i}) + (\beta_1 + b_{1i}) x_{ij} + \varepsilon_{ij}, \tag{4.2}
$$

where we assume that

$$
b_i \sim N(0, \Psi)\ ,\ \varepsilon_{ij} \sim N(0, \sigma^2)
$$

where $b_i = [b_{0i}, b_{1i}]$. However, it is clear from Figure 4.1 that there is limited variation of the slope among the data sets, so inclusion of a second random effect parameter is unlikely to significantly improve the individual fits. In cases where the parameter relations are more complex, it may not be immediately obvious if the inclusion of a particular random effect is necessary or if parameters will be identifiable in the sense that they can be uniquely determined from data. As we will detail after the next example, techniques to isolate identifiable parameters in fixed effect models are often ineffective for mixed-effects models, thus motivating the algorithms presented here.

As a second example, we employ a nonlinearly parameterized mixed-effects model. We consider the spring equation

$$
\begin{aligned}
& m \frac{d^2 y}{dt^2} + k y = 0 \\
& y(0) = y_0\ ,\ \frac{dy}{dt}(0) = 0,
\end{aligned}
\tag{4.3}
$$

which has the solution

$$
y(t) = y_0 \cos\left(t \sqrt{k/m}\right). \tag{4.4}
$$

When conducting numerical experiments, we assume that the observed initial displacements are

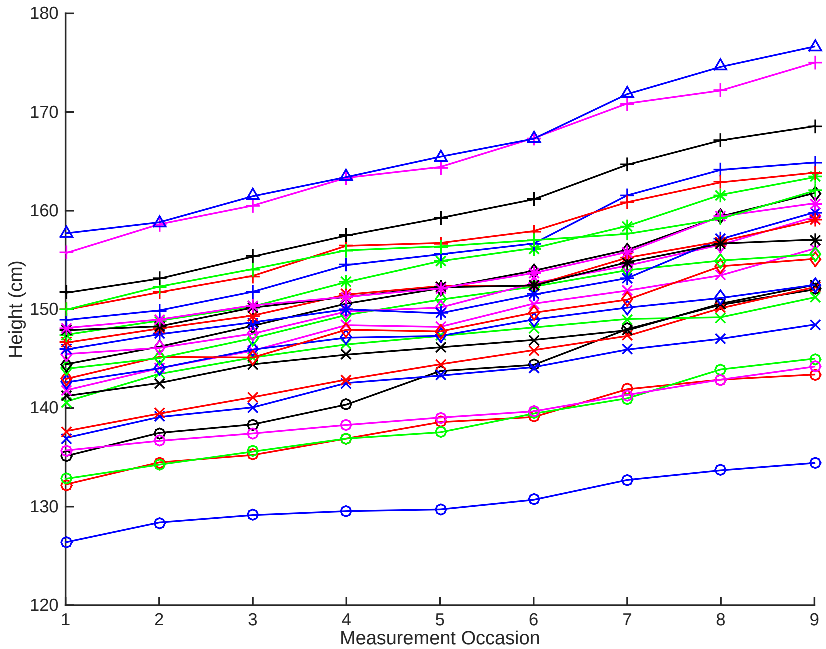


Figure 4.1: Height measurements in centimeters for 26 boys on nine occasions from [50].

drawn from a population and thus can be represented as $y_0 + b_i$ where $b_i \sim N(0, \Psi)$. The resulting statistical model is

$$y_{ij} = f(t_{ij}; \beta, b_i) + \varepsilon_{ij},$$

where y_{ij} is the measured spring displacement at time t_{ij} for the ith experiment and $\varepsilon_{ij} \sim N(0, \sigma^2)$ are the measurement errors. The mathematical model is

$$f(x_{ij}; \beta, b_i) = (y_0 + b_i)\cos\left(t\sqrt{k/m}\right),$$

where $\beta = [m, k, y_0]$ are the fixed effect parameters. We first note that mass m and stiffness k are not jointly identifiable since they appear in the solution as a quotient. Furthermore, additional random effects would not be identifiable for data generated in this manner.

To motivate issues that must be addressed when considering parameter subset selection techniques for mixed-effects models, we first detail the concepts of identifiable and influential parameters. Here we consider general models of the form $y = f(q)$.

The parameters $q = [q_1, \ldots, q_p]$ are identifiable at q^* if $f(q) = f(q^*)$ implies that $q = q^*$ for all admissible $q \in \mathcal{Q}$, where $\mathcal{Q}$ is the admissible space of parameters. The parameters q are identifiable with respect to a space $I(q)$, termed the identifiable subspace, if this holds for a $q^* \in I(q)$. Readers are referred to Chapter 6 of [37] for details regarding the specification of $I(q)$ for the spring model (4.3).

Influential parameter spaces are sometimes defined differently in various disciplines. We define the parameters q to be noninfluential on the space $NI(q)$ if $|f(q)-f(q^*)| < \varepsilon$ for all q and $q^* \in NI(q)$. The space of influential parameters, $I(q)$, is defined to be the orthogonal complement of $NI(q)$ with respect to the admissible parameter space q.

Because noninfluential parameters yield model responses that vary minimally over $NI(q)$, it is common to ascertain noninfluential parameters by testing whether local sensitivities satisfy the property $\frac{\partial y}{\partial q_i} \approx 0$. More generally, one can employ global sensitivity techniques to rank the influence of the parameters. For example, variance-based techniques, such as Sobol' analysis, rank the degree to which response uncertainties can be apportioned to input uncertainties. Morris indices provide quasi-global parameter rankings by averaging coarse finite-difference approximations computed at randomly-chosen points in the parameter space [37].

Whereas these techniques often prove very effective for fixed effect models, they are generally much less effective for mixed-effects models since they do not distinguish between the global nature of the fixed effects and the local nature of the random effects. Since standard sensitivity methods are generally ineffective, model selection techniques are frequently employed to determine insensitive parameters, which can be fixed without greatly affecting the model response.

Many model selection techniques for mixed-effects models utilize information criteria, or selection scores generated by minimizing a penalized least squares error function for various model versions [6, 9, 29]. The most common choices for information criteria include the Akaike Information Criteria (AIC), the Bayesian Information Criteria (BIC), and modified versions of these such as the marginal AIC (mAIC) and the conditional AIC (cAIC) [29]. The use of information criteria is beneficial in that it can be applied to both linear and nonlinear mixed-effects models [6, 29]. However, use of information criteria can be prohibitive due to the number of model candidates that need to be considered [3, 29].

Typically, model selection starts with a version of a mixed-effects model containing p_F fixed effects and p_R random effects, and it attempts to reduce the number of parameters to avoid overfitting. This means that the model candidates are generated with all possible combinations of each of the $p_F + p_R$ parameters being used as a variable or fixed as an appropriate constant. Hence, $2^{p_F+p_R}$ possible models must be considered. Some efforts have been made to reduce the number of tested models, including the Extended GIC (EGIC) and the Restricted Information Criteria, both of which perform model “pre-selection" based on either the mean or the covariance [3]. However, this only reduces the number of models to $2^{p_F} + 2^{p_R}$, which can still be prohibitive with high-dimensional problems.

For linear mixed-effects models, the two most notable model selection alternatives to information criteria are fence methods and shrinkage methods. Like the use of information criteria, fence methods can also be computationally intensive, especially for large models [29]. Shrinkage methods, however, are a popular choice for model selection when $p_F + p_R$ is large. These methods, including mixed-effects variations of the LASSO (Least Absolute Shrinkage and Selection Operator) such as in [3], utilize a regularized least squares estimation that drives parameters to zero; those parameters that are shrunk to zero are eliminated from the model. There have been some attempts to generalize LASSO techniques to nonlinear mixed-effects models, as detailed in [31]. However, these techniques are limited in that they cannot be used for certain types of covariate relations such as power models [31].

We propose a method of model selection, which relies on parameter subset selection, that can be used for both linear and nonlinear mixed-effects models. We first introduce the parameter subset selection algorithm for mixed-effects models in Section 4.2. Then, in Section 4.3 we use the algorithm to limit the number of model candidates, reducing the computational cost of model selection via information criteria. Employing of our version of parameter subset selection, which is based on standard errors, lowers the number of model candidates from $2^{p_F+p_R}$ to $p_F + p_R$.

4.2 Parameter Subset Selection (PSS) Algorithm

Consider the general mixed-effects model

$$y_{ij} = f(x_{ij}; \beta, b_i) + \varepsilon_{ij}\ ,\ i = 1, \ldots, M\ ,\ j = 1, \ldots, n_i,$$

where, for each individual i, y_{ij} is the jth observation, x_{ij} is the jth vector of independent variables, β is the vector of fixed effect parameters, b_i represents the random effects, and ε_{ij} is the measurement error. We assume that

$$b_i \sim \mathcal{N}(0, \Psi)\ ,\ \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I),$$

where Ψ is the covariance matrix of the random effects and σ^2 is the variance of the measurement errors. Note that there are M individual data sets. Let p_F denote the number of fixed effects and p_R be the number of random effects.

To calibrate the mixed-effects model, the parameters to be estimated are the p_F components of the fixed effects vector β, the p_R components each of the random effects vectors b_i for $i = 1, \ldots, M$, and the components of the $p_R \times p_R$ random effects covariance matrix Ψ. We build upon the parameter subset selection (PSS) method developed in [8] and described in [4] and [45] to establish influential parameters for mixed-effects models. Specifically, our PSS Algorithm 6 determines the set of the $n_p \leq p = p_F + p_R$ most influential parameters from among the fixed and random effects.

Algorithm 6 Parameter Subset Selection (PSS)

1. Check the identifiability of the parameters for the purely fixed effects model. This can be done using a variety of techniques, including computing Pearson correlation coefficients, analyzing pairwise scatter plots of parameter realizations, and constructing joint parameter probability density functions using Bayesian techniques [37, 45]. Fix parameters as necessary to eliminate identifiability problems.

2. Construct an estimate of the error variance $\hat{s}^2 = \frac{\hat{R}^T\hat{R}}{N-p}$ where $\hat{R} = [\hat{R}_1, \ldots, \hat{R}_M]^T$ is the column vector of residuals with

$$b_i = \begin{bmatrix} y_{i1}(x_{i1};\hat{q}_i) - \hat{y}_{i1} \\ \vdots \\ y_{in_i}(x_{in_i};\hat{q}_i) - \hat{y}_{in_i} \end{bmatrix} = \begin{bmatrix} f(x_{i1};\hat{q}_i) - \hat{y}_{i1} \\ \vdots \\ f(x_{in_i};\hat{q}_i) - \hat{y}_{in_i} \end{bmatrix}.$$

Here $\hat{y}_{ij}$ is the jth observed model response for the ith data set, $N = \sum_{i=1}^{M} n_i$ is the total number of observations for all data sets, and $\hat{q}_i = [\hat{\beta}, \hat{r}_i]$ is the optimized parameter vector for the ith data set.

3. Using local sensitivity matrix

$$\chi(\hat{q}) = \begin{bmatrix} \chi_1(\hat{q}_1) \\ \vdots \\ \chi_M(\hat{q}_M) \end{bmatrix},$$

where

$$\chi_i(\hat{q}_i) = \begin{bmatrix} \frac{\partial y}{\partial \beta_1}(x_{i1};\hat{q}_i) & \ldots & \frac{\partial y}{\partial \beta_{p_F}}(x_{i1};\hat{q}_i) & \frac{\partial y}{\partial b_{1i}}(x_{i1};\hat{q}_i) & \ldots & \frac{\partial y}{\partial r_{p_R i}}(x_{i1};\hat{q}_i) \\ \vdots & & \vdots & \vdots & & \vdots \\ \frac{\partial y}{\partial \beta_1}(x_{in_i};\hat{q}_i) & \ldots & \frac{\partial y}{\partial \beta_{p_F}}(x_{in_i};\hat{q}_i) & \frac{\partial y}{\partial b_{1i}}(x_{in_i};\hat{q}_i) & \ldots & \frac{\partial y}{\partial r_{p_R i}}(x_{in_i};\hat{q}_i) \end{bmatrix},$$

construct an estimate of the covariance matrix $\mathrm{Cov} = \hat{s}^2\left(\chi(\hat{q})^{\mathrm{T}}\chi(\hat{q})\right)^{\dagger}$ containing the variances and correlations of the fixed and random effects. Here † denotes the Moore-Penrose pseudoinverse.

4. Determine standard errors $SE_k = \sqrt{\mathrm{Cov}(k,k)}$.

5. Calculate selection scores for all i data sets. For the kth parameter in the ith data set, the selection score is $\alpha_{k_i} = \left|SE_k/\hat{q}_{k_i}\right|$. The n_p smallest selection scores for the ith data set correspond to the data set's n_p most significant parameters.

(Continued on the next page)

Algorithm 6 Parameter Subset Selection (PSS) (continued)

6. To determine the n_p most significant parameters over all M data sets, we assign the kth parameter in the ith data set a selection index γ_{k_i}. For the most significant parameter in the ith data set—i.e., the parameter with the lowest α_{k_i} for all $k_i = 1,\ldots,p$—we set the selection index equal to one. For the next significant parameter with the next highest α_{k_i}, we assign a selection index of 2. We continue until the least significant parameter is assigned a selection index of p.
7. Calculate the selection index sums $\Gamma_{k_i} = \sum_{i=1}^{M} \gamma_{k_i}$ for all $k = 1,\ldots,p$ parameters. The n_p smallest values of Γ_{k_i} correspond to the n_p most significant parameters for all data sets. If selection index sums are equal for two parameters, we compare the selection scores of the two parameters for each of the M data sets. The parameter that most frequently has the lower selection score of the two is determined to be more significant over all M data sets.

Note that the elements of the random effects covariance matrix Ψ are not considered in parameter selection. To construct an estimate of the parameter covariance matrix utilizing asymptotic theory (Step 3) and obtain the resulting standard errors (Step 4), we need to be able to take partial derivatives of the model response with respect to the parameters of interest; the parameters of Ψ are not included in the formula for the model response. However, when we employ PSS to aid model selection, we indirectly determine the dimensions of Ψ by determining the number of random effects included in the selected model.

4.2.1 Examples Illustrating the PSS Algorithm

In this section, we illustrate the PSS Algorithm for linearly and nonlinearly parameterized models.

Example 1: Linearly Parameterized Model

We utilize an example from [3] to examine the effectiveness of the PSS algorithm for linear problems. We first generate synthetic data for 30 individuals from the true model

$$\hat{y}_{ij} = \beta_1 x_{ij1} + \beta_2 x_{ij2} + b_{1i} + b_{2i} z_{ij2} + b_{3i} z_{ij3} + \varepsilon_{ij}\ ,\ \varepsilon_{ij} \sim \mathcal{N}(0,1). \tag{4.5}$$

Here we have $n_j = 5$ observations and $j = 1,\ldots 5$ for all $i = 1,\ldots,30$. The covariates x_{ij1}, x_{ij2}, and $z_{ij\ell}$ for $\ell = 1,2,3$ are drawn from the uniform distribution $\mathcal{U}(-2,2)$, the random effects $b_i = [b_{1i}, b_{2i}, b_{3i}]$

are drawn from $\mathcal{N}(0,\Psi)$ with

$$\Psi = \begin{bmatrix} 9 & 4.8 & 0.6 \\ 4.8 & 4 & 1 \\ 0.6 & 1 & 1 \end{bmatrix},$$

and $\beta_1 = \beta_2 = 1$.

We assume that the synthetic data was quantified using nine fixed effects $\beta_1, \ldots, \beta_9$ and four random effects $b_{1i}, \ldots, b_{4i}$ via the model

$$y_{ij} = \beta_1 x_{ij1} + \beta_2 x_{ij2} + \cdots + \beta_9 x_{ij9} + b_{1i} + b_{2i} z_{ij2} + b_{3i} z_{ij3} + b_{4i} z_{ij4} + \varepsilon_{ij} \tag{4.6}$$

with unbiased measurement error $\varepsilon_{ij} \sim N(0,\sigma^2)$ for fixed but unknown error variance σ^2. With 5 observations each, we have $j = 1, \ldots, 5$ for all 30 data sets. The covariates x_{ijk} for $k = 1,2$ and $z_{ij\ell}$ for $\ell = 2,3$ were drawn from $\mathcal{U}(-2,2)$ with the generation of the synthetic data. We draw additional covariates x_{ijk} for $k = 3, \ldots, 9$ and $z_{ij\ell}$ for $\ell = 4$ from $\mathcal{U}(-2,2)$ to obtain a full set of covariates for the model (4.6), specifically x_{ijk} for $k = 1, \ldots, 9$ and $z_{ij\ell}$ for $\ell = 1,2,3$.

Since the assumed model contains more parameters than (4.5), the model (4.6) is overfitting the data, and we would expect the five most significant parameters of (4.6) to be the five parameters used to generate the data, namely β_1, β_2, b_{1i}, b_{2i}, and b_{3i}. We can rewrite (4.6) as

$$\begin{aligned} y_i &= X_i\beta + Z_i b_i + \varepsilon_i\,,\ i = 1, \ldots, 30 \\ b_i &\sim \mathcal{N}(0,\Psi)\,,\ \varepsilon_i \sim \mathcal{N}(0,\sigma^2 I). \end{aligned}$$

Here $X_i = \begin{bmatrix} x_{i1} & \ldots & x_{i9} \end{bmatrix}$ is a 5×9 matrix for $i = 1, \ldots, 30$ with x_{ik} denoting the 5×1 column vector of covariates x_{ijk} for $j = 1, \ldots 5$ and $k = 1, \ldots 9$, and $Z_i = \begin{bmatrix} 1_{j\times 1} & z_{i2} & z_{i3} & z_{i4} \end{bmatrix}$ is a 5×4 matrix for $i = 1, \ldots, 30$ with $1_{5\times 1}$ representing a column vector of ones and $z_{i\ell}$ representing the 5×1 column vector of covariates z_{ijk} for $j = 1, \ldots, 5$ and $\ell = 2,3,4$. Note that each covariate vector corresponds to only one parameter, indicating that the parameters of (4.6) are identifiable.

With the synthetic data $\hat{y}_i$ generated from (4.5), we constructed an estimate of the covariance matrix in the following manner. We used the MATLAB Statistics Toolbox function `fitlmematrix` to obtain optimal parameter estimates $\hat{q} = [\hat{q}_1, \ldots, \hat{q}_{30}]$ where $\hat{q}_i = [\hat{\beta}_1, \ldots, \hat{\beta}_9, \hat{b}_{1i}, \ldots, \hat{b}_{4i}]$ for $i = 1, \ldots, 30$. We computed the estimated error variance using the relation

$$\hat{s}^2 = \frac{\hat{R}^T \hat{R}}{N - p}$$

where

$$\hat{R} = \begin{bmatrix} \hat{R}_1 \\ \vdots \\ \hat{R}_{30} \end{bmatrix} = \begin{bmatrix} y_1(X_1, Z_1; \hat{q}_1) - \hat{y}_1 \\ \vdots \\ y_{30}(X_{30}, Z_{30}; \hat{q}_{30}) - \hat{y}_{30} \end{bmatrix}$$

is the column vector of residuals, $N = 150$ is the total number of observations for all 30 data sets, and $p = p_F + p_R = 13$ is the total number of fixed and random effects parameters. The covariance estimate was then calculated as

$$\text{Cov}(\hat{q}) = \hat{s}^2(X^T X)^{-1},$$

where

$$X = \begin{bmatrix} X_1 & Z_1 \\ X_2 & Z_2 \\ \vdots & \vdots \\ X_{30} & Z_{30} \end{bmatrix}.$$

Note that the pseudoinverse is not necessary since $X^T X$ is full rank. We used the diagonal elements of this covariance matrix to calculate the selection scores for each parameter in each data set.

The selection scores are presented in Table 4.1. To determine the overall ordering of parameter significance, we used the selection scores to calculate the selection index sums, which are given in Table 4.2. Based on the selection index sums, we found that the parameters b_{2i}, β_2, b_{3i}, β_1, and b_{1i} are the five most significant parameters for (4.6). Hence, we recovered the parameters used in the true model, which supports the validity of the mixed-effects PSS algorithm.

Example 2: Nonlinearly Parameterized Model

In this example, we use synthetic data to verify the effectiveness of the PSS algorithm for a nonlinearly parameterized problem. We consider the classic orange tree growth model

$$y_{ij} = \frac{\beta_1 + b_{1i}}{\left(1 + e^{-[t_{ij} - (\beta_2 + b_{2i})]/(\beta_3 + b_{3i})}\right)} + \varepsilon_{ij} \tag{4.7}$$
$$b_i \sim \mathcal{N}(0, \Psi)\,,\ \varepsilon_i \sim \mathcal{N}(0, \sigma^2 I)$$

from [24] where, for the jth data point in the ith data set, y_{ij} is the tree circumference in millimeters, t_{ij} is the time in days, $\beta = [\beta_1, \beta_2, \beta_3]$ are the fixed effects, and $b_i = [b_{1i}, b_{2i}, b_{3i}]$ are the random effects. We generated pairwise plots for the fixed effect parameters of (4.7) using the Delayed Rejection Adaptive Metropolis (DRAM) algorithm [15, 37]. Observing these pairwise plots in Figure

Table 4.1: Selection scores for (4.6) for all 30 data sets.

Data Set	β_1	β_2	β_3	β_4	β_5	β_6	β_7	β_8	β_9	b_{1i}	b_{2i}	b_{3i}	b_{4i}
1	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.62	0.051	0.37	13.6
2	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.24	0.026	0.035	0.95
3	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.041	0.58	0.023	2.42
4	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.14	0.025	0.079	11.9
5	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.056	0.042	0.055	3.05
6	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.099	0.070	0.031	1.30
7	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.11	0.007	0.010	0.25
8	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.089	0.035	0.049	2.96
9	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.55	0.010	0.026	0.49
10	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.042	0.025	0.016	0.71
11	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.055	0.043	0.080	1.76
12	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.058	0.035	0.041	10.3
13	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.36	0.026	0.061	1.54
14	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.25	0.021	0.038	1.01
15	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.053	0.014	0.014	0.48
16	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.554	0.046	1.162	22.9
17	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.280	0.031	0.51	3.85
18	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.079	0.006	0.015	0.26
19	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.075	0.016	0.019	0.62
20	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.046	0.012	0.014	0.52
21	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.069	0.086	0.036	4.02
22	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.38	0.011	0.017	0.40
23	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.065	0.030	0.019	0.76
24	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.159	0.014	0.023	0.56
25	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	1.28	0.012	0.023	0.49
26	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.080	0.012	0.58	1.04
27	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.149	0.020	0.077	0.95
28	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.34	0.12	0.088	1.63
29	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.11	0.15	0.054	3.77
30	0.051	0.047	1.21	0.46	0.91	2.02	0.43	0.31	0.22	0.039	0.017	0.032	2.54

Table 4.2: Selection index sums for the linear mixed-effects model (4.6).

Γ_{β_1}	Γ_{β_2}	Γ_{β_3}	Γ_{β_4}	Γ_{β_5}	Γ_{β_6}	Γ_{β_7}	Γ_{β_8}	Γ_{β_9}	$\Gamma_{b_{1i}}$	$\Gamma_{b_{2i}}$	$\Gamma_{b_{3i}}$	$\Gamma_{b_{4i}}$
107	76	344	265	309	379	235	200	165	164	56	95	335

4.2, we determined that that none of the three plots show single-valuedness; thus, the fixed effect parameters are identifiable.

We generated synthetic data for 30 individuals using the model

$$\hat{y}_{ij} = \frac{\beta_1 + b_{1i}}{\left(1 + e^{-[t_{ij} - (\beta_2 + b_{2i})]/\beta_3}\right)} + \varepsilon_{ij}\ ,\ \varepsilon_{ij} \sim \mathcal{N}(0,1), \tag{4.8}$$

where

$$t_i = \begin{bmatrix} 118 & 484 & 664 & 1004 & 1231 & 1372 & 1582 \end{bmatrix}^T.$$

Here t_i is the 1×7 column vector of time covariates for all $i = 1, \ldots, 30$, $b_i = [b_{1i}, b_{2i}]$ are drawn from $\mathcal{N}(0, \Psi)$ with

$$\Psi = \begin{bmatrix} 15 & 0 \\ 0 & 30 \end{bmatrix},$$

and $\beta_1 = 175$, $\beta_2 = 800$, and $\beta_3 = 300$. Note that (4.8) does not contain an b_{3i} random effect. Thus, using the mixed-effect PSS algorithm with the synthetic data, we would expect to find that b_{3i} is the least influential parameter of (4.7).

Using the 30 synthetic data sets $\hat{y}_i$, we employed the MATLAB Statisitics Toolbox function `nlmefit` to obtain optimal parameter estimates $\hat{q} = [\hat{q}_1, \ldots, \hat{q}_{30}]$ where $\hat{q}_i = [\hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3, \hat{b}_{1i}, \hat{b}_{2i}, \hat{b}_{3i}]$

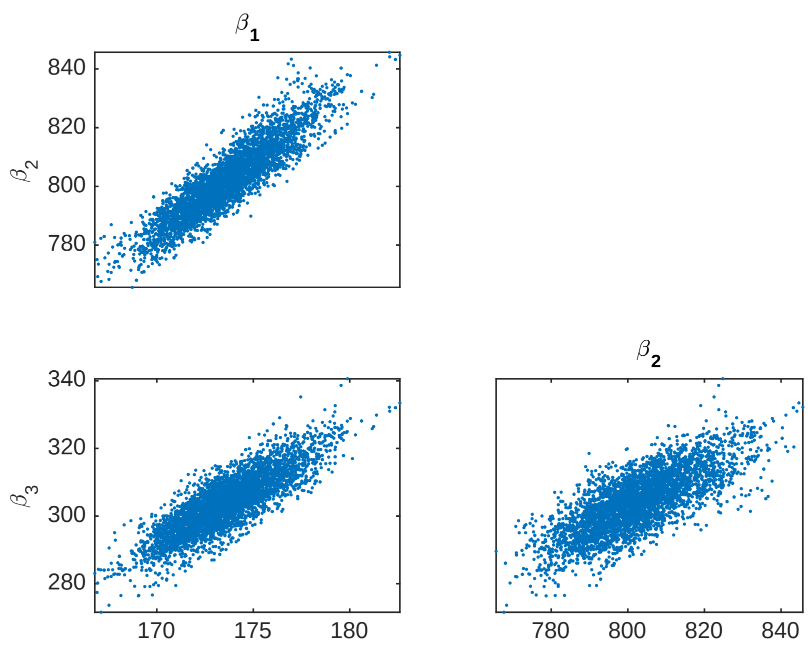


Figure 4.2: Pairwise plots generated by DRAM for the fixed effects of (4.7).

for $i = 1, \ldots, 30$. We computed the estimated variance

$$\hat{s}^2 = \frac{\hat{R}^T \hat{R}}{N - p}$$

where

$$\hat{R} = \begin{bmatrix} \hat{R}_1 \\ \vdots \\ \hat{R}_{30} \end{bmatrix} = \begin{bmatrix} y_1(t_1; \hat{q}_1) - \hat{y}_1 \\ \vdots \\ y_{30}(t_{30}; \hat{q}_{30}) - \hat{y}_{30} \end{bmatrix}$$

is the column vector of residuals, $N = 210$ is the total number of observations for all 30 data sets, and $p = p_F + p_R = 6$ is the total number of fixed and random effects parameters. We then calculated the covariance matrix

$$\mathrm{Cov}(\hat{q}) = \hat{s}^2 (\chi(\hat{q})^T \chi(\hat{q}))^{\dagger}$$

where † is the Moore-Penrose pseudoinverse and

$$\chi(\hat{q}) = \begin{bmatrix} \chi_1(\hat{q}_1) \\ \vdots \\ \chi_{30}(\hat{q}_{30}) \end{bmatrix}.$$

For $i = 1, \ldots, 30$, we computed the local sensitivities

$$\chi_i(\hat{q}_i) = \left[\frac{\partial y}{\partial \beta_1}(t_i; \hat{q}_i) \quad \frac{\partial y}{\partial \beta_2}(t_i; \hat{q}_i) \quad \frac{\partial y}{\partial \beta_3}(t_i; \hat{q}_i) \quad \frac{\partial y}{\partial b_{1i}}(t_i; \hat{q}_i) \quad \frac{\partial y}{\partial b_{2i}}(t_i; \hat{q}_i) \quad \frac{\partial y}{\partial b_{3i}}(t_i; \hat{q}_i) \right]$$

evaluated at the optimized parameter vector $\hat{q}_i = [\hat{\beta}_1, \hat{\beta}_2, \hat{\beta}_3, \hat{b}_{1i}, \hat{b}_{2i}, \hat{b}_{3i}]$ for the ith data set, containing 7 data points. Note that we use the pseudoinverse because $\chi(\hat{q})^T \chi(\hat{q})$ has rank 3 and is, therefore, non-invertible.

Using the diagonal elements of the estimated covariance matrix, we calculated the selection scores for each of the six parameters for all 30 data sets. The selection scores are tabulated in Table 4.3, and the resulting selection index sums are given in Table 4.4. The selection index sums indicate that b_{3i} is overall the least significant parameter. This is expected since the synthetic data was generated from (4.8), which did not include an b_{3i} random effect. Hence, the results support the validity of the mixed-effects version of the PSS algorithm when applied to the benchmark nonlinearly parameterized problem.

Table 4.3: Selection scores for the nonlinear mixed-effects orange tree circumference model (4.7).

Data Set	$\lvert SE_1/\hat{\beta}_1\rvert$	$\lvert SE_2/\hat{\beta}_2\rvert$	$\lvert SE_3/\hat{\beta}_3\rvert$	$\lvert SE_4/\hat{b}_{1i}\rvert$	$\lvert SE_5/\hat{b}_{2i}\rvert$	$\lvert SE_6/\hat{b}_{3i}\rvert$
1	0.6479	0.8960	0.7948	71.88	107.87	299.06
2	0.6479	0.8960	0.7948	120.71	95.67	286.26
3	0.6479	0.8960	0.7948	933.49	16.62	1443.12
4	0.6479	0.8960	0.7948	43.45	1413.06	15274.02
5	0.6479	0.8960	0.7948	73.24	1099.03	389.92
6	0.6479	0.8960	0.7948	60.12	239.68	903.57
7	0.6479	0.8960	0.7948	17.44	52.88	276.29
8	0.6479	0.8960	0.7948	241.53	124.86	543.32
9	0.6479	0.8960	0.7948	37.08	76.50	300.59
10	0.6479	0.8960	0.7948	20.75	373.93	667.00
11	0.6479	0.8960	0.7948	8.700	103.18	279.28
12	0.6479	0.8960	0.7948	32.33	62.07	1810.03
13	0.6479	0.8960	0.7948	26.03	374.38	575.21
14	0.6479	0.8960	0.7948	26.70	44.37	315.69
15	0.6479	0.8960	0.7948	29.69	150.75	283.38
16	0.6479	0.8960	0.7948	29.02	388.47	2454.32
17	0.6479	0.8960	0.7948	57.87	93.86	362.46
18	0.6479	0.8960	0.7948	37.02	290.18	441.05
19	0.6479	0.8960	0.7948	269.34	154.94	1602.34
20	0.6479	0.8960	0.7948	162.72	205.70	998.84
21	0.6479	0.8960	0.7948	31.49	130.49	285.46
22	0.6479	0.8960	0.7948	515.65	65.19	1341.84
23	0.6479	0.8960	0.7948	68.15	107.18	309.76
24	0.6479	0.8960	0.7948	92.36	559.78	338.74
25	0.6479	0.8960	0.7948	14806.29	338.39	373.06
26	0.6479	0.8960	0.7948	23.32	191.33	213.74
27	0.6479	0.8960	0.7948	173.02	48.49	222.54
28	0.6479	0.8960	0.7948	33.50	85.21	672.02
29	0.6479	0.8960	0.7948	70.99	193.41	1336.61
30	0.6479	0.8960	0.7948	35.48	320.27	396.94

Table 4.4: Selection index sums for the nonlinear mixed-effects orange tree circumference model (4.7).

Γ_{β_1}	Γ_{β_2}	Γ_{β_3}	$\Gamma_{b_{1i}}$	$\Gamma_{b_{2i}}$	$\Gamma_{b_{3i}}$
30	90	60	128	145	177

4.3 Model Selection

The PSS algorithm ranks the fixed and random effects of a mixed-effects model in order of significance and defines a subset of the n_p most influential parameters for $1 \le n_p \le p$ where $p = p_F + p_R$. These results allow us to limit the number of models considered for testing via information criteria. Instead of testing all $2^{p_F + p_R}$ models as is commonly the case, we only test the p models derived from the subsets of the $n_p = 1, 2, \ldots, p$ most significant parameters. We apply this method of model selection to the previous models to verify the PSS algorithm.

For the linear model (4.6), we used both AIC and BIC scores to select from among the 13 models corresponding to the parameter subsets generated by the PSS algorithm. For 200 trials, we used (4.5) to generate 30 individual data sets with 5 observations each and performed model selection. In Table 4.5, we list the percentage of the trials for which the PSS-aided model selection determined the correct model, the correct fixed effects, and the correct random effects. Moreover, Table 4.5 compares the results from the PSS-aided model selection to those from various other techniques as reported by [3]. Note that for the M-ALASSO method, the label of AIC or BIC denotes the method used for tuning. For REML.IC and EGIC, it denotes the method used for pre-selection. The methods denoted as LASSO, ALASSO, and Stepwise utilize REML.IC with either AIC or BIC for pre-selection and then the listed shrinkage or stepwise technique for the final step of model selection. The performance of the PSS-aided model selection via BIC compares favorably to the other methods, and while the PSS-aided AIC does not do quite as well, it out-performs the other techniques that make use of the AIC.

We similarly applied our model selection technique to the nonlinear model (4.7), calculating the AIC and BIC scores for the 6 models determined by the PSS algorithm. For 200 trials, we employed the model (4.8) to generate 30 individual data sets with 7 observations each and calculated the AIC and BIC scores for the appropriate models. In Table 4.6, we tabulate the percentage of the trials for which the PSS-aided model selection determined the correct model, the correct fixed effects, and the correct random effects. For the nonlinearly parameterized model, we did not compare our model selection method to the alternative techniques listed in Table 4.5 because these techniques are not easily applicable to nonlinear models. Although we do not compare our PSS-aided model selection to other methods for nonlinear mixed-effects models, note that for both the AIC and BIC, the percentages compare favorably with those for the linear model selection given in Table 4.5.

Table 4.5: Model selection results for linear model (4.6). The results from PSS-aided model selection are shown along with the results from various other methods as reported in [3].

Method		%Correct Model	%Correct Fixed Effects	% Correct Random Effects
PSS	BIC	67	75	91
PSS	AIC	26	27	91
M-ALASSO	BIC	71	73	79
EGIC	BIC	47	56	52
REML.IC	AIC	19	21	62
REML.IC	BIC	59	59	68
Stepwise	AIC	13	15	62
Stepwise	BIC	51	53	68
LASSO	AIC	17	21	62
LASSO	BIC	45	47	68
ALASSO	AIC	21	24	62
ALASSO	BIC	62	63	68

Table 4.6: Model selection results for nonlinear model (4.7) using PSS to aid in model selection.

Method		%Correct Model	%Correct Fixed Effects	% Correct Random Effects
PSS	BIC	55	100	55
PSS	AIC	62	100	62

4.4 Conclusion

Because local and global sensitivity analysis techniques are generally ineffective for mixed-effects models, one must determine alternative techniques to ascertain which parameters are noninfluential and can be effectively fixed for subsequent computations. The most common model section technique, the use of information criteria, is limited in that it can be computationally demanding for problems with large numbers of parameters. We developed a parameter subset selection algorithm for mixed-effects models, which can be used to limit the number of models required to be tested with information criteria. We verified the performance of the mixed-effects PSS algorithm and successfully applied it to aid model selection.

Although the PSS algorithm performed well for the considered linear and nonlinear examples, it is important to remember that the algorithm is rooted in asymptotic theory; thus, it is vital that a sufficient data points be used to obtain accurate results. Future work will involve a more thorough exploration of this aspect of the PSS algorithm, hopefully providing more insight as to the number of observations required to trust the results of the algorithm. For additional future work, we will further test PSS-aided model selection, applying it to additional examples—both linear and nonlinear—and comparing our nonlinear model selection results to those from other techniques described in the literature. Moreover, we will explore information criteria beyond the basic AIC and BIC, which may not be ideal for use with mixed-effects models [10, 14].

CHAPTER

5 RADIATION DETECTION IN AN URBAN SETTING

Our objective in this component of the investigation is to determine the location and intensity of a radiation source in an urban environment. As a prototypical setting, we consider a 250 m × 180 m block of downtown Washington D.C. as depicted in Figure 5.1. We utilize the responses of radiation detectors to help us infer the source intensity and location. Note that we are primarily focused on gamma radiation, so the detector response will be in the form of photon counts. This problem set-up represents a classical inverse problem, and we employ Bayesian inference to estimate the location and intensity of the radiation source. We present inference strategies for both stationary and mobile sensors.

5.1 Model Derivation

In our problem, we focus on detecting ionizing radiation in the form of gamma rays. Thus, we need a mathematical description of photon transport to formulate our model. We start with several assumptions to simplify the problem. Since the detectors and the radiation source are significantly smaller than the search domain, we can simplify the problem geometry by treating them both as points. With this modification, the solid angle subtended by the detector is assumed to be very small. Thus, a photon that undergoes a scattering event, while within the solid angle, is very unlikely to

Figure 5.1: Satellite image of the problem geometry, source location, and stationary detector positions from [39].

re-emerge in the angle of detection after the event. Moreover, photons that are absorbed by the media while traveling on the path from the source to the detector will not be detected. Hence, to model the photons reaching the detector, we are interested in quantifying the uncollided flux—that is, the flux of photons along path from the source to the detector that are not absorbed or scattered.

We begin our model formulation by deriving the formula for uncollided flux. Let I denote the angular flux of a photon and let n denote the photon number density. By definition,

$$I = cn,$$

where c is the speed of light. Let $(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}, E, t)$ denote the phase space where $\mathbf{r}$ is the position vector, $\hat{\boldsymbol{\Omega}}$ is the unit vector in the direction of the photon's travel, E is the photon energy, and t is time. Now, consider an arbitrary volume V. Then, the net rate of photon change in volume V is

$$\frac{\partial}{\partial t}\left[\int_V \frac{1}{c} I(\boldsymbol{r}, \hat{\boldsymbol{\Omega}}, E, t)\, d\boldsymbol{r}\right] dE\, d\hat{\boldsymbol{\Omega}} = (\text{gains in } V) - (\text{losses in } V). \tag{5.1}$$

We consider only uncollided photons, so we have gains only as results of

1. External source
2. Photons streaming into V.

Photon losses from the volume are solely due to

3. Any interaction, which would render the photon “collided"
4. Photons streaming out of V.

Thus, we modify (5.1) to obtain

$$\frac{\partial}{\partial t}\left[\int_V \frac{1}{c} I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)d\boldsymbol{r}\right]dE\,d\hat{\boldsymbol{\Omega}} = (\text{Gain } 1+\text{Gain } 2)-(\text{Loss } 3+\text{Loss } 4). \tag{5.2}$$

Let $S(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)$ denote the rate of external source emission in $d\boldsymbol{r}$ about $\boldsymbol{r}$, $d\hat{\boldsymbol{\Omega}}$ about $\hat{\boldsymbol{\Omega}}$, etc. It then follows that

$$(\text{Gain } 1) = \int_V S(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)d\boldsymbol{r}\,dE\,d\Omega. \tag{5.3}$$

Now, the net rate at which photons stream out of V is given by

$$(\text{Loss } 4)-(\text{Gain } 2) = \int_{\mathbf{S}} d\mathbf{S}\hat{\boldsymbol{\Omega}} I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)dE\,d\hat{\boldsymbol{\Omega}},$$

where $\hat{\boldsymbol{\Omega}}$ is a unit vector. Applying the Divergence Theorem, we obtain

$$(\text{Gain } 2)-(\text{Loss } 4) = \int_{\mathbf{S}} d\mathbf{S}\,\hat{\boldsymbol{\Omega}} I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)dE\,d\hat{\boldsymbol{\Omega}} = \int_V d\boldsymbol{r}\,\hat{\boldsymbol{\Omega}}\cdot\nabla I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)dE\,d\hat{\boldsymbol{\Omega}}. \tag{5.4}$$

By definition of the total scattering cross-section Σ_t, we obtain

$$(\text{Loss } 3) = \int_V d\boldsymbol{r}\;\Sigma_t(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)dE\,d\hat{\boldsymbol{\Omega}}. \tag{5.5}$$

Substituting (5.3), (5.4), and (5.5) into (5.2) yields

$$\int_V d\boldsymbol{r}\left[\frac{1}{c}\frac{\partial}{\partial c}I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)+\hat{\boldsymbol{\Omega}}\cdot\nabla I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)+\Sigma_t(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)-S(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)\right]dE\,d\hat{\boldsymbol{\Omega}}=0.$$

Since V is arbitrary, it follows that

$$\int_V d\boldsymbol{r}\;f(\boldsymbol{r})=0 \Rightarrow f(\boldsymbol{r})=0.$$

Thus, we set

$$\frac{1}{c}\frac{\partial}{\partial c}I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)+\hat{\boldsymbol{\Omega}}\cdot\nabla I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)+\Sigma_t(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)-S(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)=0. \tag{5.6}$$

Rearranging (5.6), we obtain

$$\frac{1}{c}\frac{\partial}{\partial c}I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)+\hat{\boldsymbol{\Omega}}\cdot\nabla I(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)+\Sigma_t(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t)=S(\boldsymbol{r},\hat{\boldsymbol{\Omega}},E,t), \tag{5.7}$$

which is the transport equation for uncollided photons. Disregarding the depletion of the media and assuming that the measurement time is short enough that the source activity is constant throughout allows us to eliminate the time dependence, so we obtain

$$\hat{\boldsymbol{\Omega}}\cdot\nabla I(\mathbf{r},E,\hat{\boldsymbol{\Omega}})+\Sigma_t(\mathbf{r},E,\hat{\boldsymbol{\Omega}})I(\mathbf{r},E,\hat{\boldsymbol{\Omega}})=S(\mathbf{r},E,\hat{\boldsymbol{\Omega}}). \tag{5.8}$$

Moreover, we can assume that our source is monoenergetic, only emitting photons with energy E_0. Then, we can express the photon source as

$$S(\mathbf{r},E,\hat{\boldsymbol{\Omega}})=4\pi_0\delta(\|\mathbf{r}-\mathbf{r}_s\|)\delta(E-E_0),$$

where I_0 and $\mathbf{r}_s$ are respectively the nominal source intensity and location. Substituting this expression for $S(\mathbf{r},E,\hat{\boldsymbol{\Omega}})$ into (5.8) yields

$$\hat{\boldsymbol{\Omega}}\cdot\nabla I(\mathbf{r},E,\hat{\boldsymbol{\Omega}})+\Sigma_T(\mathbf{r},E,\hat{\boldsymbol{\Omega}})I(\mathbf{r},E,\hat{\boldsymbol{\Omega}})=4\pi_0\delta(\|\mathbf{r}-\mathbf{r}_s\|)\delta(E-E_0), \tag{5.9}$$

where δ is the Dirac delta density. Solving for the uncollided flux I, we obtain

$$I(E)=I_0\exp\left(-\int_{\mathbf{r}_d-\mathbf{r}_s}\Sigma_t\,d\mathbf{s}\right)\delta(E-E_0),$$

where $\mathbf{r}_d$ is the location of the detector and $\mathbf{r}_s$ is the location of the source.

Now, we derive the formula for the detector response. Since the detector is assumed to be small, the intensity striking its face is given by

$$I_d=\hat{\boldsymbol{\Omega}}_d I(E),$$

and we can approximate the solid angle of detection as

$$\hat{\boldsymbol{\Omega}}_d=\frac{A}{4\pi\|\mathbf{r}_d-\mathbf{r}_s\|_2^2},$$

where A is the area of the detector. Hence, we can quantify the number of photons from the source counted by the detector as

$$\Gamma_s(E) = I(E)\Delta\varepsilon_{int}\frac{A}{4\pi\|\mathbf{r}_d - \mathbf{r}_s\|_2^2},$$

where Δ is the dwell time and ε_{int} is the detector efficiency. Integrating over all possible energies, we obtain

$$\Gamma_s = I_0\Delta\varepsilon_{int}\frac{A}{4\pi\|\mathbf{r}_d - \mathbf{r}_s\|_2^2}\exp\left(-\int_{\mathbf{r}_d-\mathbf{r}_s}\Sigma_t\, d\mathbf{s}\right).$$

Assuming the presence of only a single source, we can account for the total detector response by including background radiation B to obtain

$$\Gamma = I_0\Delta\varepsilon_{int}\frac{A}{4\pi\|\mathbf{r}_d - \mathbf{r}_s\|_2^2}\exp\left(-\int_{\mathbf{r}_d-\mathbf{r}_s}\Sigma_t\, d\mathbf{s}\right) + B.$$

Thus, in the context of the inverse problem with parameters $\mathbf{r}_s$ and I_0, the detector response is given by

$$\Gamma(I_0, \mathbf{r}_s) = I_0\Delta\varepsilon_{int}\frac{A}{4\pi\|\mathbf{r}_d - \mathbf{r}_s\|_2^2}\exp\left(-\int_{\mathbf{r}_d-\mathbf{r}_s}\Sigma_t\, d\mathbf{s}\right) + B. \tag{5.10}$$

5.2 Data Generation

Let Σ_t denote the total nuclear cross section of a material. We assume that the air has $\Sigma_t = 0$, and we denote the cross sections of the N_b buildings as Σ_t^k for $k = 1, \ldots, N_b$. We assume that the buildings are homogeneous and have a constant Σ_t throughout. We draw the Σ_t for each of the buildings from a scaled uniform distribution. For a building measuring 25 meters in length, we draw $\Sigma_t \sim \mathscr{U}(0.5, 1.5)$. Similarly, for a building 50 meters in length, we draw $\Sigma_t \sim \mathscr{U}(2, 3)$. Letting ε_{int} be the intrinsic efficiency of the radiation detector, we set $\varepsilon_{int} = 0.62$. Let B indicate the detected background radiation. We set $B = 300$ cps. Let the position of the source and the N_d detectors be respectively denoted by $\mathbf{r}_s$ and $\left\{\mathbf{r}_d^j\right\}_{j=1}^{N_d}$. Let A be the surface area of the detector. Here, we simulate use of a 3 inch × 3 inch NaI detector. Let Δ be the dwell time—that is, the time for which the detectors make measurements. We assume the same dwell time for all of the detectors. We take ten consecutive measurements each with a ten-second dwell time. Given Σ_t^k, ε_{int}, B, A, $\mathbf{r}_s$, $\left\{\mathbf{r}_d^j\right\}_{j=1}^{N_d}$, I_0, and Δ, we generate one vector of measurements $\hat{\Gamma} = [\hat{\Gamma}_1, \ldots, \hat{\Gamma}_{N_d}]^T$ with the following steps.

1. Set I_0 to the nominal intensity of the simulated radiation source. We employ a source of 1 mg of Cs-137. Note that the nominal intensity is 3.214×10^{12} for 1 g of Cs-137, so we set $I_0 = 3.214 \times 10^9$ Bq.

2. Ray-trace from $\mathbf{r}_s$ to $\mathbf{r}_{d_j}$ to obtain path lengths $\gamma_{1j}, \ldots, \gamma_{m_j j}$. The path lengths indicate the distances that the radiation from the source travels through the air before hitting either a

building or the detector. That is, the distance D_j from the source to the jth detector is split into m_j path lengths due to $m_j - 1$ buildings partitioning the ray from the source to the detector. Thus, we have $D_j = \sum_{i=1}^{m_j} \gamma_{ij}$. We employ the Python package Shapely to perform the ray tracing.

3. Obtain the value Γ_j, denoting the number of particles arriving at the detector, with the formula

$$\Gamma_j = I_0 \Delta \varepsilon_{int} \frac{A}{4\pi \|\mathbf{r}_{d_j} - \mathbf{r}_s\|_2^2} \exp\left(-\sum_{i=1}^{m_j} \Sigma_t^i \gamma_{ij}\right) + B.$$

4. Draw $\hat{\Gamma}_j \sim \text{Poisson}(\Gamma_j)$. We draw from the distribution ten times to reflect the ten consecutive data collections with dwell time $\Delta = 1$s.

5.3 Methods: DRAM and DREAM

For the purpose of verification, we employed two Bayesian parameter estimation techniques: the Delayed Rejection Adaptive Metropolis (DRAM) algorithm and the DiffeRential Evolution Adaptive Metropolis (DREAM) algorithm. As described in Chapter 2, the DRAM algorithm [15, 37] is a modified version of the Metropolis-Hastings algorithm, a Markov Chain Monte Carlo (MCMC) technique used to randomly sample from probability distributions. Since radiation count data is Poisson distributed, we employ a Poisson likelihood function. Hence, we provide a more general version of DRAM in Algorithm 7 for which the notation does not rely on a normal likelihood function. We still use the Delayed Rejection component given in Algorithm 3.

While the adaptation and delayed rejection components of DRAM are often sufficient for obtaining posterior parameter densities, there are some types of problems for which the algorithm is not efficient, especially those involving complex, multimodal, or heavy-tailed posteriors [37]. In response to these concerns, parallel chains were incorporated into adaptive Metropolis algorithms, resulting in differential evolution Markov chain methods. One such method is the DREAM algorithm described in [48]. The steps of DREAM are detailed in Algorithm 8.

Algorithm 7 Delayed Rejection Adaptive Metropolis for a General Likelihood Function [15, 37]

(1) Set design parameters s_p and k_0 and the number of chain iterates M.

(2) Determine q^0 =arg $\min_q \sum_{i=1}^n [v_i - f_i(q)]^2$.

(3) Set V_0 equal to the initial covariance estimate and set $R_0 = \text{chol}(V_0)$.

(4) For $k = 1, ..., M$

  (a) Sample $z_k \sim \mathcal{N}(0, I)$.

  (b) Construct candidate $q^* = q^{k-1} + R_{k-1}^T z_k$.
  Note that this is equivalent to sampling $q^* \sim N(q^{k-1}, V_{k-1})$.

  (c) Sample $u_\alpha \sim \mathcal{U}(0,1)$.

  (d) Compute $\alpha(q^*|q^{k-1}) = \min\left(1, \dfrac{\pi(y|q^*)\pi_0(q^*)}{\pi(y|q^{k-1})\pi_0(q^{k-1})}\right)$ using likelihood function π and prior π_0.

  (e) If $u_\alpha < \alpha$,

    Set $q^k = q^*$.

  else

    Enter Delayed Rejection Algorithm 3.

  endif

  (f) If $\mod(k, k_0) = 1$,

    Update $V_k = s_p \text{cov}(q^0, q^1, ..., q^k)$ and $R_k = \text{chol}(V_k)$.

  else

    $V_k = V_{k-1}$.

  endif

Algorithm 8 DREAM from [48]

Let the current state of the chain, corresponding to the ith chain iteration, be given by the p-dimensional vector q_i. Let the jth element of the current state of the ith chain be given by q_j^i.

1. Let p be the number of parameters and define p' as the number of parameters that are jointly updated. Set $p' = p$. Set the number of chains N and the number of pairs δ. Define $\gamma(\delta, p')$ as the number of randomly sampled pairs. We use $\gamma(\delta, p') = 2.38/\sqrt{2\delta d}$.

2. Draw an initial population—that is, a vector of p parameters for each of the N chains denoted by $\{q^i,\ i = 1, 2, .., N\}$—using the prior distributions for each parameter.

3. For $i = 1, \ldots, N$

   (a) Generate a new candidate using the proposal function

$$q^{*i} = q^i + \left(I_p + E\right)\gamma(\delta, p')\left[\sum_{k=1}^{\delta} q^{r_1(k)} - \sum_{\ell=1}^{\delta} q^{r_2(\ell)}\right] + \varepsilon, \tag{5.11}$$

   where $r_1(k),\ r_2(\ell) \in \{1, 2, .., N\}$ with $r_1(k) \neq r_2(\ell) \neq i$, for $k, \ell = 1, 2, .., \delta$. Here each entry of the $p \times p$ matrix E is drawn from $\mathscr{U}(-b, b)$ and the vector ε is drawn from $\mathscr{N}(0, b^*)$ where $|b| < 1$ and b and b^* are smaller than the variance of the posterior density.

   (b) Since in many cases it is not optimal to update all p dimensions simultaneously, DREAM employs randomized subspace sampling, updating each dimension with probability CR and decreasing p' accordingly. Here

$$q_j^{*i} = \begin{cases} q_j^i, \text{ if } u \leq 1 - CR\ ,\ p' = p' - 1 \\ q_j^{*i}, \text{ otherwise}\ , \end{cases} \tag{5.12}$$

   where $j = 1, \ldots, p$ and u is drawn from uniform distribution $\mathscr{U}(0, 1)$. Note that if we have crossover probability $CR = 1$, all dimensions are updated and $p' = p$.

   (c) Compute Metropolis acceptance probability

$$\alpha(q^{*i}) = \min\left[1, \frac{\pi(\nu|q^{*i}) \cdot \pi_0(q^{*i})}{\pi(\nu|q^i) \cdot \pi_0(q^i)}\right].$$

   (d) Sample $u \in \mathscr{U}(0, 1)$.

   If $\alpha > u$,

   Set $q^{i+1} = q^{*i}$.

   else

   Set $q^{i+1} = q^i$.

   endif.

(Continued on the next page)

Algorithm 8 DREAM (continued)

4. Remove possible outlier chains using inter-quartile range (IQR) statistics during the burn-in period.

5. Compute $\hat{R}_j$, the Gelman-Rubin convergence diagnostic [13], for all p dimensions with $j = 1, \ldots, p$ using the last 50% of each chain.

6. If $\hat{R}_j < 1.2$, which indicates that the chains have converged,

   End the algorithm.

   else

   Return to chain evolution in Step 3.

We applied the Delayed Rejection Adaptive Metropolis (DRAM) algorithm to estimate the parameters x, y, and I_0 employing synthetic data generated as described in Section 5.2 using $N_d = 10$ detectors along with the building geometry, source location, and detector positions shown in Figure 5.1. We used the ordinary least squares estimates as the starting values for each of the parameter chains. The x and y coordinates were bound based upon the limits of the geometry. The I_0 parameter was bound by the interval $[5 \times 10^8, 5 \times 10^{10}]$. For all three parameters, we utilized flat priors constrained by the specified bounds. We employed the Poisson log likelihood function

$$\ell(\mathbf{r}_s, I_0 | \hat{\Gamma}) = \sum_{j=1}^{N_d} \left[\left(\sum_{i=1}^{10} \hat{\Gamma}_{ji} \log\big(\Gamma_j(I_0, \mathbf{r}_s)\big) \right) - 10 \cdot \Gamma_j(I_0, \mathbf{r}_s) \right], \tag{5.13}$$

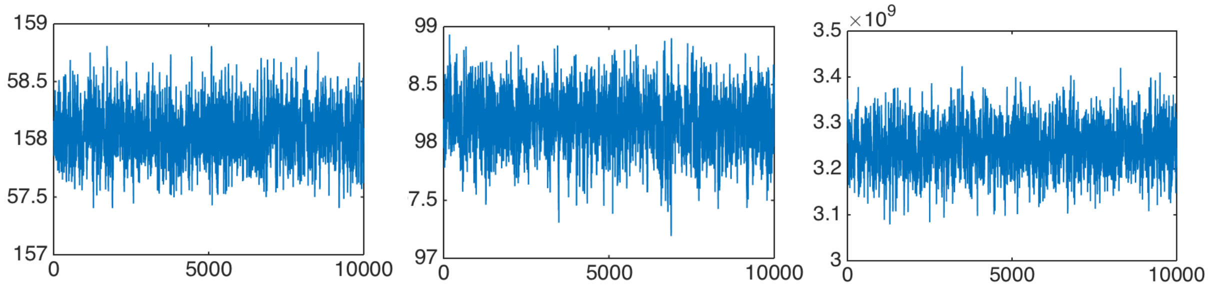
where Γ_j is the model response from (5.10) for the jth detector and $\hat{\Gamma}_{ji}$ is the ith component of the synthetic data vector $\hat{\Gamma}_j$ for the jth detector.

After a burn-in period of 3000, we reran the code for 10^4 iterations. The resulting chains are shown in Figure 5.2. Visual inspection of the chains suggests that they are burned in, and the Geweke diagnostic values reported in Table 5.1 further support this. Using the mean chain values as our parameter estimates, we obtained $\hat{x} = 158.06$, $\hat{y} = 98.19$, and $\hat{I}_0 = 3.249 \times 10^9$, which compare favorably with the parameter values used to generate the synthetic data: $x = 158$, $y = 98$, and $I_0 = 3.214 \times 10^9$.

To verify our results obtained using Bayesian inference via DRAM, we also estimated the parameters x, y, and I_0 using the DREAM algorithm. For DREAM, we utilized the Poisson likelihood (6.2) as well as uniform priors bounded by the constraints described in the previous section for the DRAM algorithm. We employed ten chains of length 10^4 for each parameter, utilizing a total of 10^5 function evaluations. The starting values for each of the ten chains were drawn from the respective

Table 5.1: Numerical results from DRAM using the Poisson likelihood (6.2). The reported parameter estimates are the mean chain values.

	Parameter Estimates	Geweke Diagnostic
$\hat{x}$	158.06	0.99962
$\hat{y}$	98.188	0.99953
$\hat{I}_0$	3.249×10^9	0.99966



(a) (b) (c)

Figure 5.2: Chains generated by DRAM for source characteristics (a) x, (b) y, and (c) I_0.

prior distributions of each parameter.

Figure 5.3 shows the plot of the ten chains for all three parameters. The truncated chains in Figure 5.3 (d)–(f) suggest that the chains have burned in after the 2000$^{\text{th}}$ sample. This is confirmed by the plots of the Gelman-Rubin R-statistic in Figure 5.4. For the parameter estimates, we used the mean value of the final 25% of the chains, which are comprised of samples from the stationary posterior distributions. The resulting parameter estimates $\hat{x} = 158.05$, $\hat{y} = 98.18$, and $\hat{I}_0 = 3.251 \times 10^9$ compare favorably with the true parameter values. As shown in Table 5.2, the parameter estimates produced by DREAM and DRAM agree. Moreover, as shown in Figure 5.5, the probability density functions (pdf's) resulting from the two methods appear to be nearly identical. While the pdf's are visually very similar, there are quantitative options for determining their agreement. The pdf's were constructed using parameter chains, which—after the burn-in period—are simply samples from the posterior distribution. We can use energy statistics to test the hypothesis that the samples in the DREAM and DRAM chains for a given parameter come from the same distribution [41, 42, 43]. This constitutes future work.

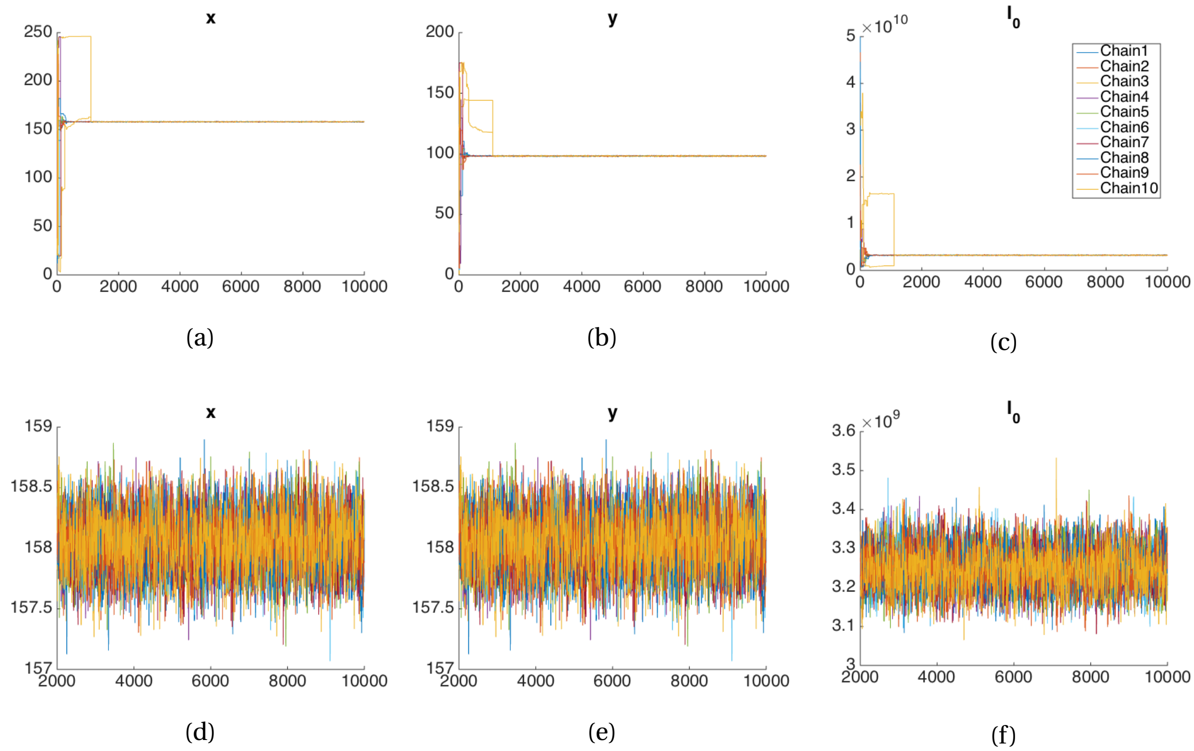


Figure 5.3: Full DREAM chains for (a) x, (b) y, and (c) I_0. Truncated DREAM chains only including the burned-in portion for (d) x, (e) y, and (f) I_0.

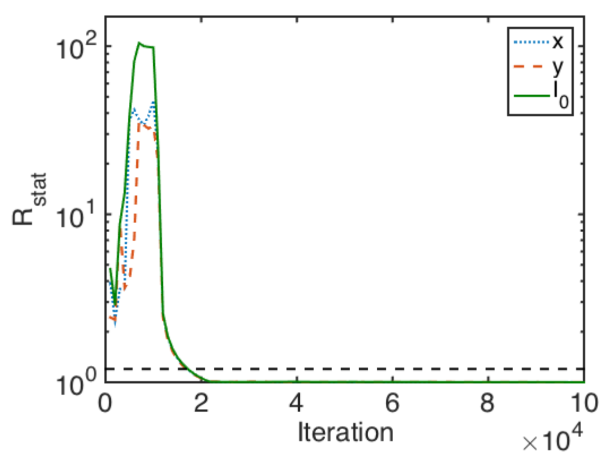


Figure 5.4: Gelman-Rubin R-statistic at each DREAM chain iteration. R-statistic values below 1.2 suggest that the chain has converged to its stationary distribution.

Table 5.2: Numerical results from DRAM and DREAM using the Poisson likelihood (6.2) along with the true values used to generate the synthetic data. The reported parameter estimates are the mean chain values.

Parameter	DRAM	DREAM	True
$\hat{x}$	158.06	158.05	58
$\hat{y}$	98.188	98.18	98
$\hat{I}_0$	3.249×10^9	3.251×10^9	3.214×10^9

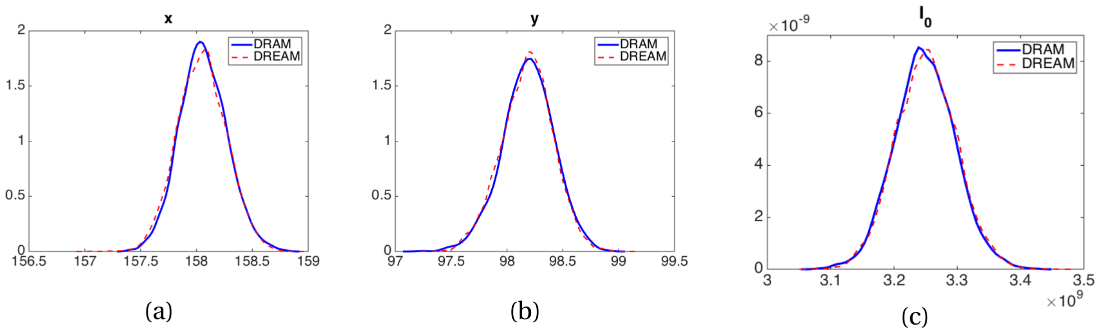


Figure 5.5: Comparison of marginal pdf's for source components (a) x, (b) y, and (c) I_0 obtained with DRAM and DREAM.

5.4 Optimal Mobile Sensor Deployment via Mutual Information

So far in this chapter, we have only considered radiation detection via stationary sensors. Moreover, these stationary detectors were randomly positioned with no attention to the optimality of their placement aside from a cursory visual inspection that they were well-dispersed throughout the domain. The optimal placement of stationary detectors is a challenging problem in the context of source localization because ideal detector placement is highly problem dependent. In particular, optimal detector placement is based on the location of the source, which will vary among problems and generally be unknown *a priori*. As an alternative, mobile sensors can easily adapt to differing source locations when paired with a movement strategy that places the sensors in optimal measurement locations. Here we employ mutual information, a dimensionless value that measures how much one random variables informs another, to guide mobile sensor movement.

5.4.1 Employing Mutual Information to Choose an Experimental Design Conditions for Optimal Model Calibration

In a variety of problems, low-fidelity models are used in place of full physics models that are computationally prohibitive. We can calibrate these low-fidelity models using experimental data, but running experiments to obtain this data can be difficult or costly. In such cases where obtaining data is expensive, it is desirable to limit the number of experiments and, for the experiments that are carried out, to provide the maximum amount of information. Thus, given a current data set, we seek an experimental design condition producing a measured response that optimally informs the low-fidelity code. That is, we wish to obtain a new data point that provides the greatest reduction in the uncertainty of the low-fidelity parameter estimates, and we seek the design condition for the experiment that provides such response data. In this section, we describe the use of mutual information to choose such an optimal design condition within a Bayesian framework. This process is based on the work in [5, 18, 44] and is more generally related to the work in [19].

Given a set of experimental observations $D_{n-1} = \{\tilde{d}_1, \tilde{d}_2, ..., \tilde{d}_{n-1}\}$, we seek a design condition $\xi_n^* \in \Xi$, which would generate new data point $(\xi_n^*, \tilde{d}_n)$, such that we optimally reduce the uncertainty in the low-fidelity model parameters $q \in \mathbb{R}^p$ when we re-calibrate the model using the new set of observations $D_n = \{\tilde{d}_n, D_{n-1}\}$. We use mutual information to choose ξ_n^* from the set of possible design conditions Ξ. Note that we can use Bayes' Rule to represent how the posterior parameter distributions change with the inclusion of the additional point $(\xi_n, \tilde{d}_n)$. In particular, we have

$$p(q|D_n) = \frac{p(D_n|q)p(q)}{p(D_n)} = \frac{p(\tilde{d}_n, D_{n-1}|q)p(q)}{p(\tilde{d}_n, D_{n-1})}. \tag{5.14}$$

Let d_n denote the unknown response from the yet-to-be-performed experiment under design condition ξ_n. To quantify the mutual information between d_n and parameter values q, we employ Shannon entropy estimates as in [44]. For a random variable Q with associated pdf $p(q)$ for $q \in \mathcal{Q}$, where $\mathcal{Q}$ is the parameter space, the Shannon entropy is given by

$$H(Q) = -\int_{\mathcal{Q}} p(q) \log(p(q)) dq$$

for the prior and

$$H(Q|\nu) = -\int_{\mathcal{Q}} p(q|\nu) \log(p(q|\nu)) dq$$

for the posterior distribution given observations ν. Based on our goals for the selecting a design condition, we define the utility function

$$U(d_n, \xi_n) = \int_{\mathcal{Q}} p(q|d_n, D_{n-1}) \log p(q|d_n, D_{n-1}) dq - \int_{\Omega} p(q|D_{n-1}) \log p(q|D_{n-1}) dq, \tag{5.15}$$

which quantifies the amount of information provided by the low-fidelity measurement d_n obtained under design condition $\xi_n \in \Xi$. We can then compute the average amount of information obtained with design condition ξ_n by marginalizing over the set of all unknown future observations $\mathscr{D}$ as

$$\mathbb{E}_{d_n}[U(d_n,\xi_n)] = \int_{\mathscr{D}} U(d_n,\xi_n) p(d_n|D_{n-1},\xi_n) d d_n. \tag{5.16}$$

We substitute (5.15) into (5.16) to obtain the expected utility

$$\begin{aligned}
\mathbb{E}_{d_n}[U(d_n,\xi_n)] &= \int_{\mathscr{D}}\int_{\mathscr{Q}} p(q,d_n|D_{n-1},\xi_n)\log\frac{p(q,d_n|D_{n-1},\xi_n)}{p(d_n|D_{n-1},\xi_n)} dq\, dd_n \\
&\quad -\int_{\mathscr{D}}\int_{\mathscr{Q}} p(q,d_n|D_{n-1},\xi_n)\log p(q|D_{n-1}) dq\, dd_n \\
&= \int_{\mathscr{D}}\int_{\mathscr{Q}} p(q,d_n|D_{n-1},\xi_n)\log\frac{p(q,d_n|D_{n-1},\xi_n)}{p(q|D_{n-1})p(d_n|D_{n-1},\xi_n)} dq\, dd_n \\
&= I(q;d_n|D_{n-1},\xi_n). \qquad (5.17)
\end{aligned}$$

As indicated by the notation in (5.17), the expected utility quantifies the mutual information $I(q;d_n|D_{n-1},\xi_n)$ between the low-fidelity model parameters q and unknown measurement d_n at design condition ξ_n. The optimal design condition ξ_n^* maximizes the mutual information; that is,

$$\xi_n^* = \arg\max_{\xi_n\in\Xi} I(q;d_n|D_{n-1},\xi_n).$$

We then perform an experiment under the optimal condition ξ_n^* and use the resulting observation $\tilde{d}_n$ to recalibrate the model parameters q. If design replication is not desired, the chosen ξ_n^* is then eliminated from the design set Ξ. Note that the integral in (5.16) generally cannot be evaluated directly; hence, numerical methods are necessary for the calculation of mutual information. In this dissertation, we utilize the kNN (k^{th}-Nearest Neighbor) method proposed in [17] to obtain our mutual information values; see Appendix A for more details. An alternative option, not employed here, for numerically calculating mutual information is the Approximate Nearest Neighbor (ANN) method [2]. Compared to the kNN method, the computational cost of the ANN algorithm is much less. Whereas kNN requires a computational time on the order of $\mathscr{O}(n^2)$, the ANN algorithm running time is on the order of $\mathscr{O}(n\log n)$ for n data points [18].

5.4.2 Mutual Information for Mobile Sensors

The problem of guiding mobile sensors is analogous to finding an optimal design condition to inform low-fidelity model parameters. The goal is to move a given sensor to the measurement location—or, design condition—that provides the most information about the source location and intensity, which are the parameters of the simplified radiation transport model (5.10). Note that we are not specifying the sensor dynamics or trajectories but rather are specifying a sequence of discrete locations where measurements optimally inform parameters. Thus, when considering mobile sensors, we take a similar approach to that described in Section 5.4.1. Based on the approach described in the previous section and the algorithm detailed in [18], we propose the mobile sensor movement strategy in Algorithm 9.

To evaluate the performance of this strategy, we once again examine the problem of determining the location and intensity of an unknown radiation source in of downtown Washington D.C. In practice, we would move to the optimal design location and obtain experimental measurements, but we are unable to collect real-life data as defined by our simulated problem. Because there are practical and ethical considerations associated with placing a radiation source in downtown Washington D.C., we are exploring other options for obtaining experimental data for similar problems. In cooperation with Oak Ridge National Lab, we have arranged a measurement campaign, which will take place at National Guard urban training center at Fort Indiantown Gap. Moreover, we are also looking into using high-fidelity codes, such the Monte Carlo N-Particle (MCNP) transport code developed by Los Alamos National Lab, to generate higher-quality synthetic data. In this case, we could use high-fidelity synthetic data to inform the low-fidelity model calibration as in [18]. However, in this dissertation, we simply generate synthetic data from the simplified model (5.10) for all possible measurement locations *a priori* and supply the synthetic measurements in place of a true measured response. As before, the synthetic response for each location is a vector of ten values, corresponding to the detector counts obtained over ten consecutive one-second measurement periods.

For the set of possible design locations Ξ, we constructed a discrete grid of possible measurement sites. As shown in Figure 5.6, we employed 29 regularly-spaced grid points, requiring that they be placed outside of the buildings. With a regular grid throughout the geometry, randomly selecting the starting locations for the three sensors is likely to result in the choice of three detector positions that register little to no signal from the radiation source. To avoid this, we intentionally selected three points that were well-dispersed throughout the geometry. Specifically, we used $(61.65, 44.08)$, $(92.48, 132.25)$, and $(184.96, 66.13)$ as the initial locations. We employed a value of $N = 5000$ and a value of $k = 6$ for the kNN algorithm. For parameter calibration via DRAM, we used the same bounds as in the stationary sensor problem. For x and y, we employed uniform priors characterized by these bounds, but for I_0, we employed a normal approximation of a Poisson prior with mean and the variance equal to 3.214×10^9. Use of a more informative prior, compared to the uniform prior in the stationary detector problem, helped to decrease the burn-in period for the multiple model calibrations. We also employed chain starting values equal to the values of x, y, and I_0 used to generate the data to further reduce the computation time.

The order in which the design conditions were selected are shown in Figure 5.7. The source is shown as a red triangle, and the three initial points are shown as green stars. The order in which the design conditions were chosen appears to be reasonable. The early points were chosen to be either

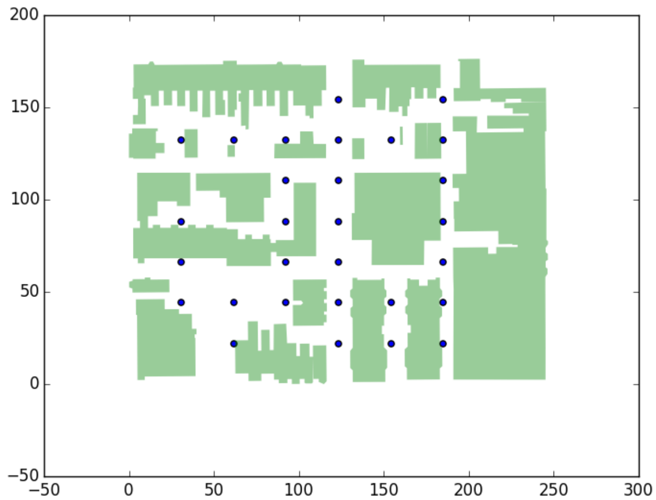


Figure 5.6: Grid of possible design locations for mobile sensors.

near the source or near the corner of a building, which suggests good exploration of the geometry. Moreover, the final measurement locations to be selected were those at the perimeter. While the search strategy seems to be valid, the results are less promising for the resulting posterior parameter distributions. For both x and y, the posteriors produced by DRAM at each iteration of Algorithm 9 were fairly uniform with the chains hitting against the enforced bounds. The chains and pdf's for the final iteration are shown in Figures 5.8 and 5.9, respectively. In this regard, the results from the stationary detectors are much better with the narrow posterior distributions suggesting a much smaller amount of uncertainty in the parameter estimates. The uniform posteriors produced by the mobile sensor measurements are likely due to the high number of design locations at which little to no signal from source is detected. In fact, the responses at two of the three initial sensor locations, $(61.6540, 44.0830)$ and $(92.4810, 132.2500)$, are close to the nominal background count of 300, indicating little to no influence from the source. A possible remedy for this problem would be to implement a filtering process which removes noninformative design locations from the set Ξ and to pair that process with a method for initializing all three detectors at informative points. This constitutes future work.

Algorithm 9 Mobile Sensor Movement Strategy

1. Set N equal to the number of samples to be used in the kNN algorithm.
2. Define the set Ξ of n_L possible measurement locations.
3. Initialize with three sensors placed at locations ξ_1, ξ_2, ξ_3 chosen from Ξ. Take readings to obtain data set $\eta_3 = [(\xi_1, \tilde{d}_1), (\xi_2, \tilde{d}_2), (\xi_3, \tilde{d}_3)]$. Note that these three initial locations should be chosen so that they are well-dispersed throughout the domain.
4. The remaining possible locations for mobile sensors are $[\xi_4, \xi_5, ..., \xi_{n_L}]$.
5. For $r = 4, \ldots, n_L - 1$,

   (a) Let Ξ_r be the remaining design conditions.

   (b) Employ DRAM as detailed in Algorithm 7 using the data set η_{r-1} to construct a $3 \times N$ matrix of parameters chains $\{q^i\}_{i=1}^N$.

   (c) Send $\{q^i\}$ to the kNN algorithm detailed in Appendix A.

   (d)The kNN algorithm returns a single design condition ξ_{n_r}, which indicates where one of the three mobile sensors should move. Move a sensor to the location, and measure the detector response to obtain $\tilde{d}_{n_r}$. Here we use synthetic data in place of the measured response. Append the new location and response $(\xi_{n_r}, \tilde{d}_{n_r})$ to data set η_{r-1} to obtain η_r.

   (e) Remove ξ_{n_r} from Ξ_r to obtain Ξ_{r+1}.

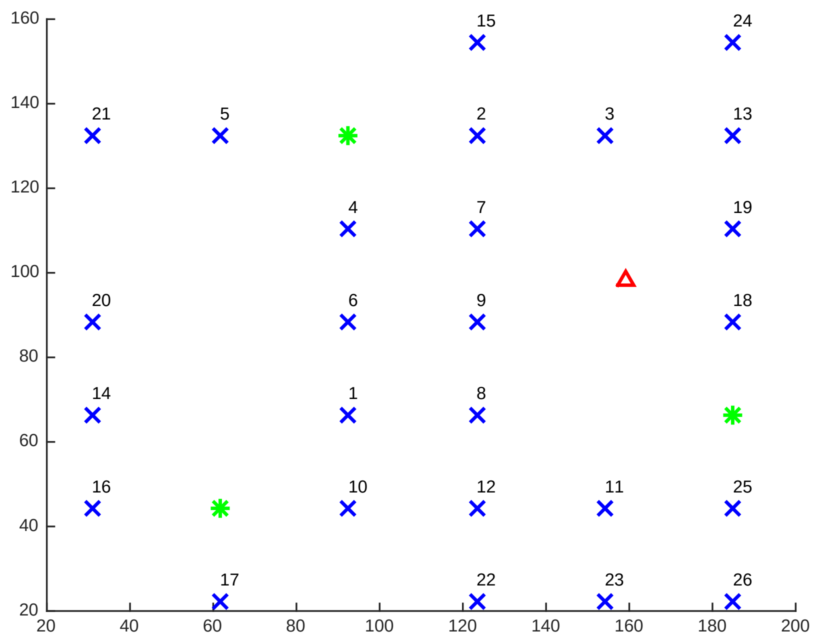


Figure 5.7: Order in which the sensor locations were employed. The blue x's indicate the possible measurement locations of Ξ, and the number indicates the iteration of Algorithm 9 for which the design location was selected. The green stars represent the original locations of the three sensors. The red triangle shows the location of the source.

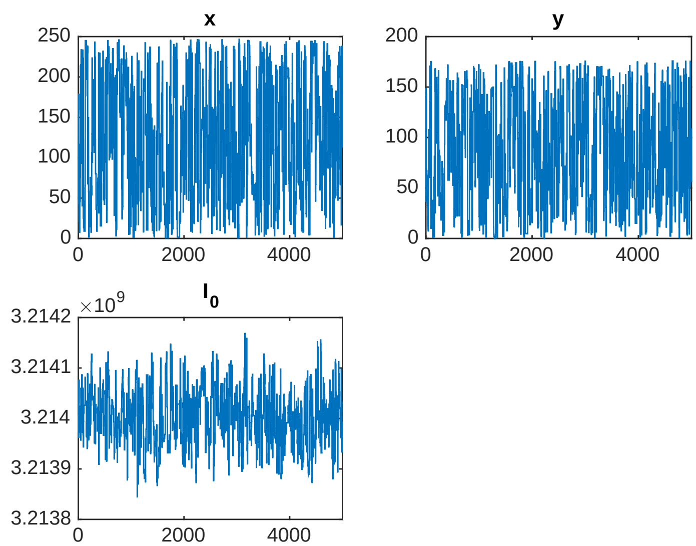


Figure 5.8: DRAM chains from the final iteration of Algorithm 9 with 25 potential design conditions.

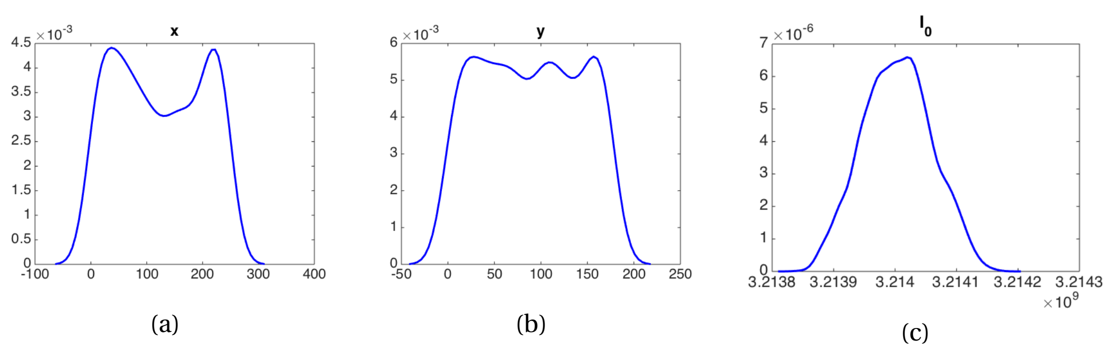


Figure 5.9: Marginal pdf's constructed from the DRAM chains of the final iteration of Algorithm 9 with 25 potential design conditions.

CHAPTER

6

MODELING RADIATION DETECTION USING MIXED-EFFECTS FOR BACKGROUND VARIATION

In Chapter 5, we employed Bayesian parameter estimation to solve a radiation source localization inverse problem. Recall that in both the model (5.10) and the generation of synthetic data, we treated background radiation as a constant set to 300 counts per second (cps). To improve the physical accuracy of the model, we opted to estimate the background radiation, treating it as a parameter. However, simple parameterization does not account for the varying background radiation at each of the detector locations. Measurements taken at the Fort Indiantown Gap National Guard Training Center in Pennsylvania indicate that background radiation fluctuates with location, even within a localized urban area. Moreover, these measurements suggest that the background radiation is normally distributed within the urban locale. To account for the underlying distribution of the background radiation, we employed mixed-effects modeling, allowing us to estimate the individual values of the background radiation at each of the detectors as well the mean and standard deviation of the underlying distribution.

6.1 Revised Model and Data Generation

Modifying (5.10) to include background variation, we obtain the model

$$\Gamma_j(I_0, \mathbf{r}_s, B_j) = I_0 \Delta\varepsilon_{int} \frac{A}{4\pi\|\mathbf{r}_{d_j} - \mathbf{r}_s\|_2^2} \exp\left(-\int_{\mathbf{r}_{d_j}-\mathbf{r}_s} \Sigma_t \, d\mathbf{s}\right) + B_j,$$

where B_j is the individual background parameter for the jth detector, $j = 1, \ldots, N_d$, and all other terms are as defined previously. We assume that

$$B_j \sim \mathcal{N}(\mu, \sigma^2).$$

We generate synthetic data as in Section 5.2 with the exception of Step 4. To obtain the value Γ_j, which denotes the number of photons counted by the detector, we now employ the formula

$$\Gamma_j = I_0 \Delta\varepsilon_{int} \frac{A}{4\pi\|\mathbf{r}_{d_j} - \mathbf{r}_s\|_2^2} \exp\left(-\sum_{i=1}^{m_j} \Sigma_t^i \gamma_{ij}\right) + B_j$$

where B_j is the background radiation for the location of the jth detector. The values B_j are drawn from the normal distribution $\mathcal{N}(259.4, 13.02^2)$, which reflects the data from Fort Indiantown gap [40]. We use the data $\hat{\Gamma} = [\hat{\Gamma}_1, \ldots, \hat{\Gamma}_{N_d}]^T$ to estimate the parameters I_0, $\mathbf{r}_s = (x, y)$, and B_j for all $j = 1, \ldots, N_d$ detectors and the hyperparameters μ and σ using the mixed-effects version of DRAM from Algorithms 4 and 5. Note that estimating B_j for each detector along with the corresponding hyperparameters is equivalent to estimating the alternative mixed-effects formulation $B + r_j$ with fixed effect $B = \mu$ and random effects $r_j \sim \mathcal{N}(0, \sigma^2)$ along with the random effects standard deviation σ. For this problem, we only have a single random effect, so we can implement mixed-effects DRAM via the MATLAB MCMC Toolbox DRAM code, which operates under the assumption that the random effects covariance matrix is diagonal, without loss of generality.

6.2 Non-Unique Optimal Parameters

We generated data using a 1-mg Cs-137 radiation source located at the coordinates $(158, 98)$. We generated ten sets of observations for ten randomly-placed detectors. We denote the ten realizations from the jth detector as

$$\hat{\Gamma}_j = [\hat{\Gamma}_{j1}, \hat{\Gamma}_{j2}, \cdots, \hat{\Gamma}_{j10}].$$

Using this data, we initially attempted to simultaneously estimate the parameters x, y, I_0, and B_j for all $j = 1, \ldots, 10$ along with hyperparameters μ and σ for the radiation model (5.10) using the modified delayed rejection adaptive metropolis (DRAM) algorithm from Algorithms 4 and 5. We used the true values of x, y, and I_0 as the starting values, and we set the initial values of $B_j = 300$ for all $j = 1, \dots, 10$. The x and y coordinates were bounded based upon the limits of the geometry—that is, the grid representing the portion of interest of downtown Washington D.C. The intensity parameter I_0 was bounded by the interval $[5 \times 10^8, 5 \times 10^{10}]$, and the B_j parameters were bounded below by zero for all $j = 1, \dots, 10$.

For both x and y, we employed a flat prior, specifically one specified by the parameter bounds. Since I_0 represents the detector counts based on the source intensity, we used a normal approximation of a Poisson distribution as the prior. That is, we set the mean and the variance equal to the true value of I_0. The prior for each B_j is determined by hyperparameters μ and σ with $B_j \sim \mathcal{N}(\mu, \sigma^2)$. We employed hyperpriors $\mu \sim \mathcal{N}(300, 20^2)$ and $\sigma \sim \text{Inv-}\chi^2(1, 20^2)$. Since $\hat{\Gamma}_j \sim \text{Poisson}(\lambda_j)$ where

$$\lambda_j = \Gamma_j(I_0, \mathbf{r}_s) = I_0 \Delta\varepsilon_{int} \frac{A}{4\pi \|\mathbf{r}_{d_j} - \mathbf{r}_s\|_2^2} \exp\left(-\sum_{i=1}^{m_j} \Sigma_t^i \gamma_{ij} \right) + B_j \tag{6.1}$$

for detectors $j = 1, \dots, N_d$, we employed the Poisson log likelihood function

$$\ell(\mathbf{r}_s, I_0 | \hat{\Gamma}) = \sum_{j=1}^{N_d} \left[\left(\sum_{i=1}^{10} \hat{\Gamma}_{ji} \log\big(\Gamma_j(I_0, \mathbf{r}_s)\big) \right) - 10 \cdot \Gamma_j(I_0, \mathbf{r}_s) \right]. \tag{6.2}$$

Note that the likelihood function excludes constant terms that do not depend on λ_j as defined by (6.1).

We employed the MCMC Toolbox DRAM code utilizing the hierarchical option `localflag = 2` for the background parameters. After a burn-in period of 2×10^4, we reran the code for 5000 iterations. The resulting chains are shown in Figure 6.1. It is clear that the chains have not burned in, but more importantly, the chains appear to have shifted away from burning in near their true values, jumping to an alternative set of optimal parameters. This suggests that the parameter set is non-identifiable in the sense that the parameter estimates are not uniquely determined by the data. Hence, the optimal parameter set is not unique, and the parameters cannot be simultaneously estimated.

6.3 Narrow Prior Distribution for the Background Parameters

When good *a priori* information about model parameters is available, the Bayesian framework is advantageous. In particular, narrow prior distributions can sometimes remedy the type of identifiability problem seen in Section 6.2. We obtain such a prior for the background parameters by performing a calibration in the absence of a source.

To simulate data for a zero-source calibration, we generated synthetic data by setting $I_0 = 0$. We again obtained ten observations for each detector with each observation corresponding to a

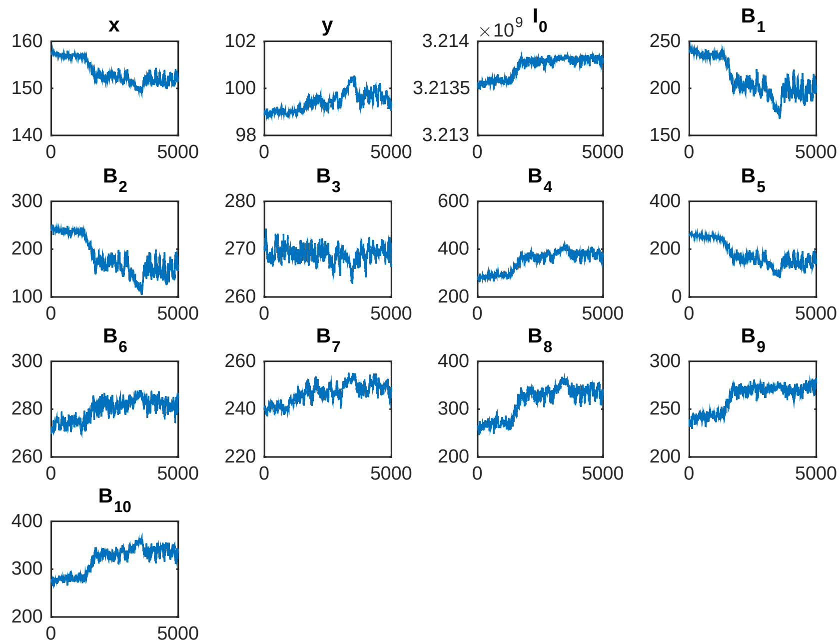


Figure 6.1: Chains generated using DRAM with Poisson likelihood (6.2) for parameters $\mathbf{r}_s = (x, y)$, I_0, and B_j for $j = 1, \ldots, 10$.

one-second dwell time. Based on (6.1), the resulting data for the jth detector is simply ten values sampled from the distribution Poisson($B_{j_{\text{true}}}$), where $B_{j_{\text{true}}}$ is the value of the background radiation at the jth detector location that was used to generate the data.

Using this synthetic data, we fixed $I_0 = 0$ and $(x, y) = (158, 98)$ in model (6.1), leaving only the background parameters B_j for $j = 1, 2, \ldots, 10$ and the hyperparameters to be estimated. Note that the choice to fix the source location as $(158, 98)$ was arbitrary since it corresponds to the location of an absent source. We employed the mixed-effects DRAM algorithm to estimate the parameters, again using the `localflag = 2` option in the MCMC Toolbox. As before, we bounded all of the background parameters to be positive. We used a burn-in period of 1.5×10^4 and final chains of length 5000. As shown in Figures 6.2 and 6.3, the chains and hyperchains have successfully burned-in. Moreover, Table 6.1 shows a comparison of the parameter and hyperparameter estimates with their true values, and the estimates closely agree with the values used to generate the data.

Note that we now have estimates for the hyperparameters μ and σ, and we can use these estimates to supply informative prior—namely, $\mathcal{N}(257.98, 14.46^2)$—for the background parameters B_J. Thus, we again attempt to simultaneously estimate the parameters of (6.1) via DRAM, this time employing the estimated values of the hyperpriors to provide a narrow prior for the estimation of the background radiation. We employ the same bounds and priors for x, y, and I_0 as those in Section 6.2, and we again bound all background parameters to be greater than zero. After a burn-in period of 3×10^4, we obtained final chains of length 5000. These parameter chains are shown in Figure 6.4. Visual inspection indicates that they have burned in. Thus, by employing a fairly tight prior on the background parameters, we prevented the chains from jumping to alternate optimal parameter values, and we were able simultaneously estimate the individual background terms along with the source location and intensity. Moreover, as shown in Table 6.2, we obtained parameter estimates close to the true values.

With our success in utilizing an informative prior with background terms to eliminate identifiability problems, a potential future direction for this problem is to resolve the components of the background radiation. The majority of radioactive decay that naturally occurs in an urban environment originates from ^{40}K, ^{238}U, and ^{232}Th. Thus, we could consider a background parameter to essentially be the sum of these three components. Even in a purely fixed effect model, dividing the background into elements, it is apparent that there would be identifiability problems without narrow priors or tight bounds on the background components. Consider such a model for a detector response

$$\Gamma(I_0, \mathbf{r}_s, B) = I_0 \Delta \varepsilon_{int} \frac{A}{4\pi \|\mathbf{r}_d - \mathbf{r}_s\|_2^2} \exp\left(- \int_{\mathbf{r}_d - \mathbf{r}_s} \Sigma_t \, d\mathbf{s} \right) + (B_K + B_U + B_T).$$

It is apparent that the optimal value of the sum $B_K + B_U + B_T$ accommodates many possibilities for the values of B_K, B_U, and B_T, but—as in the previous problem—we may be able to use a narrow prior on each of the background components so that all three may be simultaneously, and uniquely, estimated.

Table 6.1: Estimates of background radiation parameters and hyperparameters in the absence of a source obtained from the mean values of the DRAM chains.

	Parameter Estimate	True Value
B_1	250.81	244
B_2	256.06	251
B_3	270.60	268
B_4	236.33	243
B_5	255.42	250
B_6	276.51	278
B_7	240.43	240
B_8	258.43	256
B_9	259.82	258
B_{10}	251.54	260
μ	257.98	259.4
σ	14.46	13.02

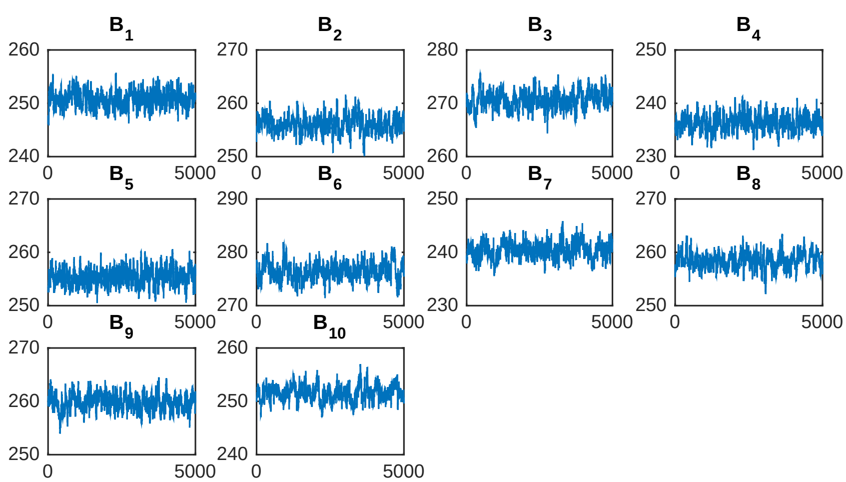


Figure 6.2: Chains generated using DRAM with Poisson likelihood (6.2) for parameters B_j for $j = 1, \ldots, 10$ in the absence of a source.

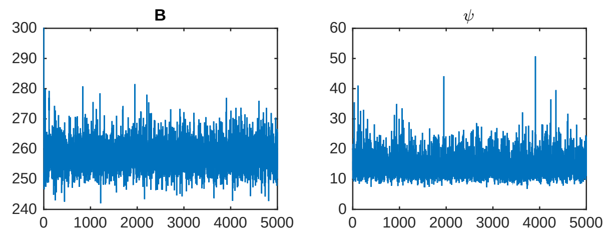


Figure 6.3: Chains generated for hyperparameters μ and σ using DRAM with Poisson likelihood (6.2) in the absence of a source.

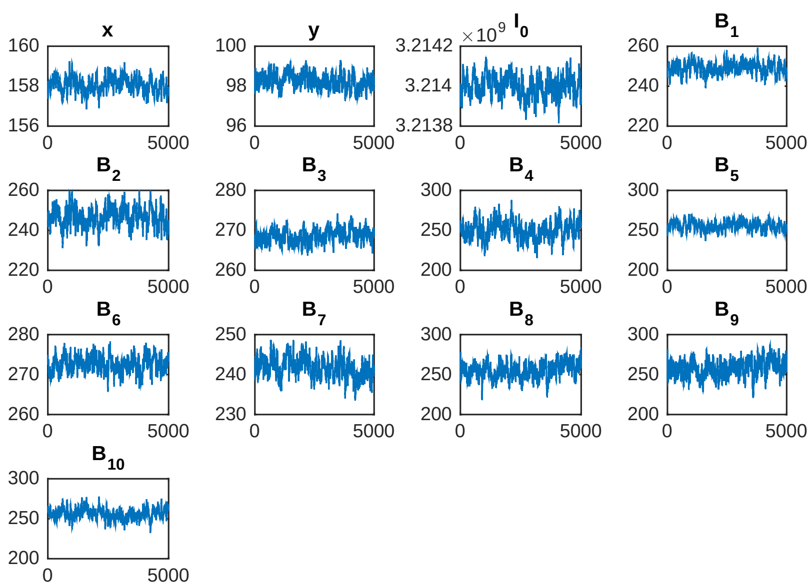


Figure 6.4: Chains generated using DRAM with Poisson likelihood (6.2) for parameters x, y, I_0, and B_j for $j = 1, \ldots, 10$ employing a narrow prior for all background terms B_j.

Table 6.2: Estimates of background and source location and intensity parameters obtained from the mean values of the DRAM chains constructed with a narrow prior on the background parameters.

	Parameter Estimate	True Value
x	158.08	158
y	98.254	98
I_0	3.214×10^9	3.214×10^9
B_1	249.11	244
B_2	246.55	251
B_3	268.52	268
B_4	250.15	243
B_5	255.81	250
B_6	272.52	278
B_7	241.76	240
B_8	255.29	256
B_9	258.18	258
B_{10}	256.22	260

CHAPTER

7 CONCLUSIONS

Whereas mixed-effects models are used throughout many areas of science, there are limited tools for performing uncertainty quantification on these types of models. In this dissertation, we introduced two novel UQ techniques tailored to mixed-effects models: a mixed-effects version of the DRAM algorithm and a parameter subset selection (PSS) algorithm. The mixed-effects DRAM algorithm allows us to perform Bayesian model calibration, obtaining accurate results even in the case of highly correlated parameters. When employed for the orange tree circumference problem, the DRAM algorithm produced an estimate of the random effects covariance matrix that was much closer to the true value than both frequentist estimation and Gibbs sampling. However, we needed to utilize a highly informative prior to obtain a good estimate of the random effects covariance matrix. Future work is needed to determine a standard approach for when good prior information is unavailable.

Our new PSS algorithm ranks the random and fixed effect parameters in order of significance, and it can be employed to aid model selection. In particular, we use our PSS algorithm to limit the number of models to be tested via information criteria. Compared to methods with similar approaches, use of the new PSS algorithm significantly lowers the number of models to be tested. Furthermore, PSS-aided model selection can be used with both linear and nonlinear mixed-effects models. Future research could involve employing new types of information criteria for model selection or implementing Morris screening techniques in the PSS algorithm to get a more global sense of parameter ranking. Also, in order to determine the relative effectiveness of PSS-aided model selection for our nonlinear example, we need to compare it to other current model selection methods. Ideally, such methods would be applicable to large dimensional problems and all types of nonlinear models.

When exploring applications for mixed-effects models in nuclear engineering, we primarily focused on locating a radiation source in an urban setting. To allow for reasonably quick model evaluations, we employed a simplified radiation transport model. We initially employed a purely fixed effect model, examining source localization strategies for both stationary and mobile sensors. For stationary sensors, we estimated the source location and intensity using both DRAM and DREAM. For mobile sensors, we used mutual information to determine which location from a set of possible measurement sites would provide the most information for parameter calibration. While our mutual information-based strategy chose the measurement sites in a logical order, the Bayesian parameter calibration at each step produced uniform posteriors for both location parameters. This is likely due to the large number of noninformative measurements sites within the possible set of locations. Further work is necessary to develop a strategy for removing the sensor locations receiving little or no signal from the source from the list of possible measurement sites.

When using the fixed effect radiation transport model, we set the background radiation to a constant 300 cps, but this is not an accurate reflection of reality. To account for varying background radiation at stationary detector locations, we employed a mixed-effects model. We attempted to use mixed-effects DRAM to calibrate the background parameters and hyperparameters along with the source location and intensity; however, the parameter chains would not burn in. Moreover, the chains appeared to be jumping away from their true location to an alternate set of optimal parameters. To remedy this nonidentifiability, we first did a calibration on the background parameters and hyperparameters in the absence of a source. Using these hyperparameter estimates, we were able to employ a tight prior on the background terms, which allowed us to simultaneously estimate the source location and intensity as well as the individual background parameters. As future work, we may be able to similarly employ tight priors to resolve the components of the background radiation, such as the radiation from ^{40}K, ^{238}U, and ^{232}Th.

REFERENCES

[1] T. Ando, *Bayesian Model Selection and Statistical Modeling*, CRC Press, Boca Raton, FL, 2010.

[2] S. Arya, D.M. Mount, N.S. Netanyahu, R. Silverman, and A.Y. Wu, "An optimal algorithm for approximate nearest neighbor searching in fixed dimensions," *Journal of the ACM*, 45(6), pp. 891–923, 1998.

[3] H.D. Bondell, A. Krishna, and S.K. Ghosh, "Joint variable selection for fixed and random effects in linear mixed-effects models," *Biometrics*, 66(4), pp. 1069–1077, 2010.

[4] H.T. Banks, A. Cintrón-Arias, and F. Kappel, "Parameter selection methods in inverse problem formulation," *Mathematical Modeling and Validation in Physiology*, volume 2064 of Lecture Notes in Math, pp. 43–73. Springer, Heidelberg, 2013.

[5] C. Bryant and G. Terejanu, "An Information-Theoretic Approach to Optimally Calibrate Approximate Models", *American Institute of Aeronautics and Astronautics*, 2012.

[6] E. Carson and C. Cobelli, *Modeling Methodology for Physiology and Medicine*, Elsevier, London, 2014.

[7] M.R. Chernick, *Bootstrap Methods: A Guide for Practitioners and Researchers*, Second Edition, John Wiley and Sons, Newtown, PA, 2008.

[8] A. Cintrón-Arias, H. T. Banks, A. Capaldi, and A. L. Lloyd. "A sensitivity matrix based methodology for inverse problem formulation," *Journal of Inverse and Ill-Posed Problems*, 17(6), pp. 545âĂŞ564, 2009.

[9] Z. Chen and D.B. Dunson, "Random effects selection in linear mixed models," *Biometrics*, 59(4), pp. 762–769, 2003.

[10] M. Delattre, M. Lavielle, and M. Poursat, "A note on BIC in mixed-effects models," *Electronic Journal of Statistics*, 8(1), pp. 456–475, 2014.

[11] B. Delyon, M. Lavielle, and E. Moulines, "Convergence of a stochastic approximation version of the EM algorithm," *The Annals of Statistics*, 27(1), pp. 94–128, 1999.

[12] J. Fox, *Applied Regression Analysis and Generalized Linear Models*, Second Edition, Sage, Thousand Oaks, CA, 2008.

[13] A.G. Gelman and D.B. Rubin, "Inference from iterative simulation using multiple sequences," *Statistical Science*, 7(4), pp. 457–511, 1992.

[14] S. Greven and T. Kneib, "On the behaviour of marginal and conditional Akaike information criteria in linear mixed models," *Biometrika* 97(4), pp. 773–789, 2009.

[15] H. Haario, M. Laine, A. Mira, and E. Saksman, "DRAM: Efficient adaptive MCMC," *Statistics and Computing*, 16(4), pp. 223–242, 2001.

[16] Z.R. Kenz, H.T. Banks, and R.C. Smith, "Comparison of frequentist and Bayesian confidence analysis methods on a viscoelastic stenosis model," *SIAM/ASA Journal of Uncertainty Quantification*, 1(1), pp. 348–369, 2013.

[17] A. Kraskov, H. Stogbauer, and P. Grassberger, "Estimating mutual information", *Physical Review E*, 69(6), 2004.

[18] A.L. Lewis, R.C. Smith, and B.J. Williams, "An information theoretic approach to use high-fidelity codes to calibrate low-fidelity codes," *Journal of Computational Physics*, Submitted.

[19] J. Liepe, S. Filippi, M. Komorowski, and M.P.H. Stumpf, "Maximizing the Information Content of Experiments in Systems Biology", *PLOS Computational Biology*, 9(1), Jan. 2013.

[20] R.J.A. Little and D.B. Rubin, *Statistical Analysis with Missing Data*, Second Edition, John Wiley and Sons, Hoboken, NJ, 2002.

[21] Mathworks, "Estimating Parameters in Linear Mixed-Effects Models," *Statistics Toolbox: User's Guide* (r2014a),http://www.mathworks.com/help/stats/estimating-parameters-in-linear-mixed-effects-models.html.

[22] Mathworks, "fitlme," *Statistics Toolbox: User's Guide* (r2014a), http://www.mathworks.com/help/stats/fitlme.html.

[23] Mathworks, "fitlmematrix," *Statistics Toolbox: User's Guide* (r2014a), http://www.mathworks.com/help/stats/fitlmematrix.html.

[24] Mathworks, "nlmefit," *Statistics Toolbox: User's Guide* (r2014a), http://www.mathworks.com/help/stats/nlmefit.html.

[25] Mathworks, "nlmefitsa," *Statistics Toolbox: User's Guide* (r2014a), http://www.mathworks.com/help/stats/nlmefitsa.html.

[26] W.H. McAdams, *Heat Transmission*, Second Edition, McGraw-Hill, New York, 1942.

[27] F.H. Morris and W.G. Whitman, "Heat transfer for oils and water in pipes," *Industrial and Engineering Chemistry*, 20(3), pp. 234–240, 1928.

[28] V. Mousseau, N. Belcourt, R. Salko, K. Ivanov, M. Avramova, and M. Doster, "Cobra-TF Parameter Exposure Work," CASL Technical Report CASL-U-2014-0198-000.

[29] S. Müller, J.L. Scealy, and A.H. Welsh, "Model selection in linear mixed models," *Statistical Science*, 28(2), pp. 135–167, 2013.

[30] J. Pinheiro and D.M. Bates, *Mixed-Effects Models in S and S-Plus*, Springer, New York, 2000.

[31] J. Ribbing, J. Nyberg, O. Caster, and E.N. Jonsson, "The lasso—a novel method for predictive covariate model building in nonlinear mixed effects models," *Journal of Pharmacokinetic Pharmacodynamics*, 34, pp. 485–517, 2007.

[32] R.K. Salko and M.N. Avramova, "COBRA-TF Subchannel Thermal-Hydraulics Code (CTF) Theory Manual," Consortium for the Advanced Simulation of Lightwater Reactors, 2015.

[33] A. Saltelli, M. Ratto, T. Andres, F. Campolongo, J. Cariboni, D. Gatelli, M. Saisana, and S. Tarantola, *Global Sensitivity Analysis: The Primer*, John Wiley and Sons, Chichester, UK, 2008.

[34] A. Saltelli, S. Tarantola, F. Campolongo, and M. Ratto, *Sensitivity Analysis in Practice: A Guide to Assessing Scientific Models*, John Wiley and Sons, Chichester, UK, 2004.

[35] K. Schmidt and R.C. Smith, "A Parameter Subset Selection Algorithm for Mixed-Effects Models," *International Journal of Uncertainty Quantification*, Submitted.

[36] G.A.F. Seber and C.J. Wild, *Nonlinear Regression*, John Wiley and Sons, Hoboken, NJ, 2003.

[37] R.C. Smith, *Uncertainty Quantification: Theory, Implementation, and Applications*, SIAM, Philadelphia, PA, 2014.

[38] G.K. Smyth, "Nonlinear regression," in *Encyclopedia of Environmetrics*, A.H. El-Shaarawi and W.W. Piegorsch, Eds., Vol 3, John Wiley and Sons, Chichester, UK, pp. 1405–1411, 2002.

[39] R. Ştefănescu, K. Schmidt, J. Hite, R. Smith, and J. Mattingly, "Hybrid optimization and Bayesian inference techniques for a non-smooth radiation detection problem," *International Journal for Numerical Methods in Engineering*, Submitted.

[40] M.W. Swinney, D.E. Peplow, A.D. Nicholson, and B.W. Patton, "NORM concentration determination in common construction materials in an urban environment," *Transactions of the American Nuclear Society*, 114, 2016.

[41] G.J. Szekely. "E-statistics: Energy of statistical sample," Technical Report, Bowling Green State University, Department of Mathematics and Statistics, 2003.

[42] G.J. Szekely and M.L. Rizzo. "Testing for equal distributions in high dimension," *InterStat*, 2004.

[43] G.J. Szekely and M.L. Rizzo. "Energy statistics: A class of statistics based on distances," *Journal of Statistical Planning and Inference*, 8(8), 2013.

[44] G. Terejanu, R.R. Upadhyay, and K. Miki, "Bayesian experimental design for the active nitridation of graphite by atomic nitrogen", *Experimental Thermal and Fluid Science*, 36, pp. 178-193, 2012.

[45] M.T. Wentworth, R.C. Smith and H.T. Banks, "Parameter selection and verification techniques based on global sensitivity analysis illustrated for an HIV model," *SIAM/ASA Journal on Uncertainty Quantification*, 4, pp 266-297, 2016.

[46] W.C. Williams, "If the Dittus and Boelter equation is really the McAdams equation, then should not the McAdams equation really be the Koo equation?," *International Journal of Heat and Mass Transfer*, 54(7), pp. 1682–1683, 2011.

[47] R.H.S. Winterton, "Where did the Dittus and Boelter equation come from?," *International Journal of Heat and Mass Transfer*," 41(4), pp. 809-810, 1998.

[48] J.A. Vrugt, C.J.F. Ter Braak, C.G.H Diks, B.A. Robinson, J.M. Hyman, and D. Higdon. "Accelerating Markov chain Monte Carlo simulation by differential evolution with self-adaptive randomized subspace sampling," *International Journal of Nonlinear Sciences and Numerical Simulation*, 10(3), pp. 273–290, 2009.

[49] J.C. Wakefield, A.F.M. Smith, A. Racine-Poon, and A.E. Gelfand, "Bayesian analysis of linear and non-linear population models using the Gibbs sampler," *Journal of the Royal Statistical Society Series C (Applied Statistics)*, 43(1), pp. 201–221, 1994.

[50] L. Wu, *Mixed Effects Models for Complex Data*, CRC Press, Boca Raton, FL, 2010.

APPENDIX

APPENDIX

A

*K*NN ALGORITHM

Since the integral in (5.16) generally cannot be evaluated directly, we employ the kNN (kth nearest neighbor) method for numerical approximation. Algorithm 10 details the procedure.

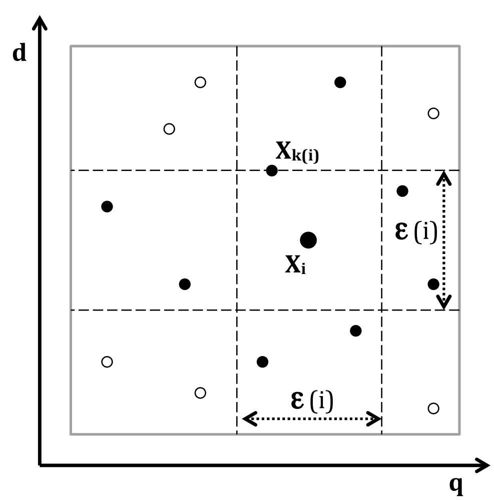


Figure A.1: Calculation of $\epsilon(i)$, $n_q(i)$, and $n_d(i)$ for the case $k = 1$ from [17, 18]. Here we illustrate $n_q(i) = 3$ and $n_d(i) = 4$.

Algorithm 10 kNN Method [17, 18]

1. Fix the value of k and define number of kNN vector elements N. We use $k = 6$ and $N = 5000$.
2. For each possible design condition $\xi_n \in \Xi$,
   (a) Let $p = \dim(q)$ be the number of parameters and let $m = \dim(d)$ be the dimension of the model output. We create a vector with $p + m$ rows and N columns.
      (i) In the first p rows, draw N samples, $\{q^i\}_{i=1}^N$, from the distribution $p(q|D_{n-1})$. For these samples, we use the DRAM chains generated for Step 5(b) in Algorithm 9.
      (ii) In the next m rows, place the $1 \times N$ low-fidelity model reponse vector $d_n(\xi_n; q_i)$, where q_i is the parameter vector from the first p rows of the ith column.
      (iii) Normalize the data vector,
      $$X = \left\{(\mathrm{diag}(s^{-1})(X_i - \mu)\right\}_{i=1}^N$$
      where $\mu = [\bar{q}, \bar{d}]^T$ is a $(p + m) \times 1$ vector of sample means and $s = [s_q, s_d]^T$ is the vector of sample standard deviations.
   (b) For each sample X_i, identify the kth nearest neighbor, $X_{k(i)}$ and compute $\epsilon(i)/2 = \|X_i - X_{k(i)}\|_\infty$.
   (c) For each sample X_i, compute $n_q(i) = \#$ points in q marginal space with at least one coordinate within distance $\epsilon(i)/2$ and $n_d(i) = \#$ points in d marginal space with at least one coordinate within distance $\epsilon(i)/2$. A visual representation of $n_q(i)$ and $n_d(i)$ is given in Figure A.1.
   (d) Estimate the mutual information as
   $$I(q; d_n|D_{n-1}, \xi_n) \approx \psi(k) - \frac{1}{N}\left[\sum_{i=1}^N \psi(n_q(i)+1) + \sum_{i=1}^N \psi(n_d(i)+1)\right] + \psi(N),$$
   where $\psi(\cdot)$ is the digamma function.
3. Use the estimated mutual information to determine ξ^*, the design such that $\max_{\xi_n \in \Xi} I(q; d_n|D_{n-1}, \xi_n) = I(q; d_n|D_{n-1}, \xi^*)$.