AN ABSTRACT OF THE DISSERTATION OF

James L. Pratt for the degree of Doctor of Philosophy in Statistics presented on November 29, 1994. Title: The Laplace Approximation and Inference in Generalized Linear Models with Two or More Random Effects.

Abstract approved: _____

Daniel Schafer

This thesis proposes an approximate maximum likelihood estimator and likelihood ratio test for parameters in a generalized linear model when two or more random effects are present. Substantial progress in parameter estimation for such models has been made with methods involving generalized least squares based on the approximate marginal mean and covariance matrix. However, tests and confidence intervals based on this approach have been limited to what is provided through asymptotic normality of estimates. The proposed solution is based on maximizing a Laplace approximation to the log-likelihood function. This approximation is remarkably accurate and has previously been demonstrated to work well for obtaining likelihood based estimates and inferences in generalized linear models with a single random effect. This thesis concentrates on extensions to the case of several random effects and the comparison of the likelihood ratio inference from this approximate likelihood analysis to the Wald-like inferences for existing estimators.

The shapes of the Laplace approximate and true log-likelihood functions are practically identical, implying that maximum likelihood estimates and likelihood ratio inferences are obtained from the Laplace approximation to the log-likelihood. Use of the Laplace approximation circumvents the need for numerical integration, which can be practically impossible to compute when there are two random effects. However, both the Laplace and exact (via numerical integration) methods require numerical optimization, a sometimes slow process, for obtaining estimates and inferences.

The proposed Laplace method for estimation and inference is demonstrated for three real (and some simulated) data sets, along with results from alternative methods which involve use of marginal means and covariances.

The Laplace approximate method and another denoted as Restricted Maximum Likelihood (REML) performed rather similarly for estimation and hypothesis testing. The REML approach produced faster analyses and was much easier to implement while the Laplace implementation provided likelihood ratio based inferences rather than those relying on asymptotic normality.

The Laplace Approximation and Inference in Generalized Linear Models with Two or More Random Effects

by

James L. Pratt

A DISSERTATION

submitted to

Oregon State University

in partial fulfillment of
the requirements for the
degree of

Doctor of Philosophy

Completed November 29, 1994
Commencement June 1995

Doctor of Philosophy dissertation of James L. Pratt presented on November 29, 1994

APPROVED:

Redacted for Privacy

Major Professor, representing Statistics

Redacted for Privacy

Chair of Department of Statistics

Redacted for Privacy

Dean of Graduate School

I understand that my dissertation will become part of the permanent collection of Oregon State University libraries. My signature below authorizes release of my dissertation to any reader upon request.

Redacted for Privacy

James L. Pratt, Author

ACKNOWLEDGEMENT

As I now sit and write this last item of this dissertation, I reflect back on those persons who contributed to the completion of this work. There are several whose impact was at least indirect, and identifying each individually would lead to a lengthy list. Those I have chosen to list herein are the ones I feel have made a significant contribution. That is not to be interpreted to mean that the efforts of others has gone unnoticed nor unappreciated by myself.

First of all, I wish to acknowledge the guidance and assistance given me by my major professor, Dr. Dan Schafer. Although the topic was not his first (nor was it his second) choice, he allowed me to pursue it and was able to help guide me when I got stuck. Dan also provided me with great help in the writing of this dissertation, often suggesting where the text flow was not ideal. Thanks.

I would also like to acknowledge the assistance from Drs. Don Pierce and Qing Liu, who provided me with preliminary results of Qing's dissertation from which my work has built upon. Much time and energy was saved in computer programming due to their input and help.

I would also like to thank the many faculty members within the Department of Statistics at Oregon State University for providing me with encouragement, the technical skills required for carrying out my research, and the opportunity to learn from them. I especially wish to thank Dr. Jutus Seely for providing me with stipends, to Drs. Ken Rowe and Cliff Pereira for their mentorship in applying my statistical training to real data, and to Dr. Dan Brunk for introducing me to the world of Statistics. I am also thankful for the help of Genevieve Downing for her great understanding and help in resolving issues related to graduate school policy, as well as carrying out simple, but time consuming tasks.

Additionally, I received great encouragement from researches who have an interest in the topic. Drs. N. Breslow, D. Boos, P. McCullagh, and C. McGilchrist each met with me at some point and discussed my results, and gave me much appreciated encouragement to continue.

While being a full-time graduate student, I had three office mates that were quite influential in encouraging me to complete this work: Danny Kugler,

Caryn Thompson and Marti McCracken. Thanks for the times together, as well as keeping me going even after I left to join the working folks.

While working in the "real" world for the past 3-plus years and still attempting to complete my dissertation, I have been fortunate enough to have had supervisors who understood the need for my completing this dissertation. These were Dr. Henry Lee (at ManTech Environmental), and Drs. Michael Meredith, Tom Filloon, John Taulbee, Eileen King, and George Jerdack (with the Biometric and Statistical Sciences department at The Procter and Gamble Co.). Each has provided me with encouragement, but more importantly with time and resources for which to complete the work. Without their support and understanding, it would have been very difficult to complete the writing.

Finally, I would like to acknowledge my family for their support and understanding over the past 7-plus years. Many, many evening and weekend hours were spent away from my wife and daughters over that span, which was at times rather difficult. I thank them for their patience and understanding for those times I couldn't spend with them. I thank them for their encouragement to keep me going. I especially thank my wife, Dee, for painstakingly checking the correctness of the table of contents, lists of figures and tables, and the bibliography. And I thank the families of both my wife's and my own for their encouragement to get this completed.

TABLE OF CONTENTS

TABLE OF CONTENTS (Continued)

LIST OF FIGURES

LIST OF FIGURES (Continued)

Figure Page

LIST OF TABLES

LIST OF TABLES (Continued)

The Laplace Approximation and Inference in Generalized Linear Models with Two or More Random Effects

1. INTRODUCTION

1.1 Purpose

This thesis proposes an approximate maximum likelihood estimator and likelihood ratio test for parameters in a generalized linear model when two or more random effects are present. In particular, attention is given to binomial and Poisson response variables arising from randomized block designs, nested designs with two levels of random nesting, split–plot designs, and certain longitudinal studies and sample surveys where it is appropriate to incorporate several random effects.

Substantial progress in parameter estimation for these models has been made with methods involving iterative generalized least squares based on the approximate marginal mean and covariance matrix of the responses (McCullagh and Nelder, 1989, Ch. 14; Goldstein, 1991) or on the conditional mean and covariance matrix of transformed responses (Gilmour, Anderson and Rae, 1985; Green, 1987; Schall, 1991; McGilchrist and Aisbett, 1991; Breslow and Clayton, 1993). These two approaches yield nearly identical estimating routines and extend the ideas for the single random effect model in Williams (1982). (See also Pierce and Sands, 1975; Breslow, 1984.) Although these estimators have been shown to yield asymptotically unbiased estimators (Liang and Zeger, 1986; Prentice, 1988; McCullagh and Nelder, 1989 Sec. 9.3) and involve relatively transparent calculations, tests and confidence intervals are limited to those based on the asymptotic normality of parameter estimators.

Approximate maximum likelihood estimators have been proposed with the aid of the EM ("expectation–maximization") algorithm (Dempster, Laird and Rubin, 1977) by treating the random effects as "missing data" (Anderson and Aitkin, 1985; Preisler, 1989). In this solution it is necessary to use numerical integration in the E–step. The value of the maximized likelihood is not calculated as a by–product of the EM calculations, and hence, computation of

likelihood ratio tests is not possible without additional numerical work. Therefore, as with the iterative generalized least squares solutions, inferences are limited to those based on the asymptotic normality of parameter estimators.

These estimators will be discussed in more detail in Chapter 2. The point for now is that inferences, in each case, are limited to those based on an assumed asymptotically normal sampling distribution. In some data problems, and with some parametrizations, there is no drawback to these inferences; but sometimes the sampling distribution of the parameter of interest is not closely approximated by a normal distribution and conclusions are potentially misleading. When a fully parametric model is specified, the likelihood ratio test is preferred.

The tradeoffs between methods based on first and second moment assumptions and those based on full parametric models are well known. Although there is a need for the methods based on the weaker assumptions when the data cannot be confirmed to fit a usable distributional model, this thesis is concerned with the extra precision of maximum likelihood estimators and the added power of the likelihood ratio test in problems where the model discussed in the following section is appropriate.

The proposed solution is based on the parameter estimates that maximize the likelihood corresponding to the marginal distribution of the response — that is, the distribution obtained by integrating out the random effects. This marginal distribution is intractable and numerical integration is cumbersome. However, a Laplace approximation to this integral is remarkably accurate. This has been used with a great deal of success in generalized linear models with one random effect by Liu and Pierce (1993). This thesis concentrates on extensions to the case of several random effects. This is important because, despite reasonable estimators mentioned above, the tests and confidence intervals based on these can be improved upon, and because numerical integration of the likelihood is considerably more difficult in the two random effects case than when there is only one random effect.

This thesis presents the estimator and tests based on the Laplace approximation, demonstrates through some theoretical and numerical results why the approximation is good, compares the tests based on this solution with existing ones, and illustrates the types of designs and samples where this procedure will be useful.

1.2 Model Definition

Let Y represent an n by 1 vector of response variables, X an n by k matrix of fixed explanatory variables, and take u_1 and u_2 to be r_1 by 1 and r_2 by 1 random vectors which are independent of each other. Suppose that

$$\mathrm{E}\{Y|u_1,u_2\} = \mu \; ; \quad g(\mu) = X\beta + A_1u_1 + A_2u_2, \tag{1.1}$$

where $g(\ )$ is a monotonic, differentiable "link" function, β is a k by 1 vector of parameters and A_1 and A_2 are known incidence matrices of dimension n by r_1 and n by r_2. Suppose further that conditional on u_1 and u_2, the components of Y are independent of one another with a one parameter exponential family distribution, and

$$u_j \sim N_{r_j}(\mathbf{0}\,,\, \sigma_j^2\boldsymbol{I}) \;,\; j = 1,\, 2 \;.$$

For the data problems considered for this dissertation, A_1 and A_2 are both full column rank. For nested random effects (as in the first two example data sets in Section 1.4), each column of A_1 will be spanned by a unique subset of columns of A_2, with A_2 often being the identity matrix. For crossed random effects (as in the third example data set in section 1.4), the columns of A_1 and A_2 are mutually orthogonal.

Two important cases are emphasized: (1) Logit regression with random effects — meaning conditional on u_1 and u_2 the components of Y are independent binomial proportions and $g(\mu) = ln[\mu/(1-\mu)]$, and (2) Poisson log–linear regression with random effects — meaning conditional on u_1 and u_2 the components of Y are independent Poisson counts and $g(\mu) = ln(\mu)$.

The assumption of normality of u_1 and u_2 has little justification and is criticized by some who feel that assumed means of 0 and constant variances are all that is required. It is then believed that results from normal linear mixed models carry over to methods using only moment assumptions and lead to more robust estimation and inferences while giving up little in precision and power. It seems natural, and mathematically convenient, that if a distributional assumption is made, that the random effects be assumed normal, as the normal

distribution is usually a good approximation for most symmetric, unimodal data. When such an assumed model's properties are understood, then comparisons to other assumptions can be made.

It is important to note that the random effects are modeled on the same scale as the fixed effects. This will be discussed more fully in Chapter 2 along with alternate models for random effects. Interest is in estimation and tests concerning the parameters β, σ_1^2, and σ_2^2. Extending to more than two random components is done by adding random components $u_3,\ldots,u_q$ in the linear predictor in a straightforward fashion.

1.3 Generality of Model (1.1)

The use of formulation (1.1) is now clarified for the following situations: randomized block experiments, nested designs, split-plot experiments, longitudinal/repeated measures studies, and sample surveys. The forms of the matrices A_1 and A_2 are given for each.

1.3.1 Randomized Block: Block and Block-by-Treatment Effects Modeled as Random

Suppose that data arise from a randomized complete block design having 5 blocks and 4 treatments. When the response distribution is binomial or Poisson the random block-by-treatment interaction is not aliased with experimental error, as it is in models with an independent variance parameter. Block and block-by-treatment interaction may be modeled as random effects. Let Y_i represent the ith component of the response vector, which has mean μ_i. Let $X_{ij} = 1$ if treatment j was applied to the ith experimental unit ($j = 1, \ldots, 4$), 0 otherwise; and let $A_{ik} = 1$ if the ith experimental unit belongs to block k ($k = 1, \ldots, 5$), 0 otherwise. Then a possible model is

$$\begin{aligned} g(\mu_i) = \quad & \beta_1 X_{i1} + \beta_2 X_{i2} + \beta_3 X_{i3} + \beta_4 X_{i4} + \\ & A_{i1}u_{1,1} + A_{i2}u_{1,2} + A_{i3}u_{1,3} + A_{i4}u_{1,4} + A_{i5}u_{1,5} + \\ & X_{i1}A_{i1}u_{2,1} + X_{i2}A_{i1}u_{2,2} + \cdots + X_{i4}A_{i5}u_{2,20} \end{aligned}$$

where the $u_{1,j}$ are independent and identically distributed as $N(0, \sigma_1^2)$ and the $u_{2,j}$ are independent and identically distributed as $N(0, \sigma_2^2)$, along with the independence of the u_1's and u_2's. The parameter σ_1^2 represents block variation, and σ_2^2 represents the variation of block-by-treatment interaction. Notice that σ_2^2 can also be thought of as a term representing extra-binomial or extra-Poisson variation on units within any block.

Suppose the data are organized such that the first 4 observations come from treatments 1, 2, 3, and 4 (respectively) applied within experimental units from block 1, the second 4 come from treatments 1, 2, 3, and 4 applied within experimental units from block 2, and so one. In this case, X may be a 20×4 design matrix of the form

$$X = \begin{bmatrix} 1 & & & \\ & 1 & & \\ & & 1 & \\ 1 & & & \\ & 1 & & \\ & & 1 & \\ & \vdots & & \end{bmatrix} = \mathbf{1}_5 \otimes \mathbf{I}_4 \ ,$$

where $\otimes$ denotes a Kronecker product, $\mathbf{1}_5$ denotes a 5×1 vector of 1's, and $\mathbf{I}_4$ denotes the 4×4 identity matrix. The incidence matrix for the 5 random block effects, A_1, is a 20×5 matrix of the form

$$A_1 = \begin{bmatrix} 1111 & & & \\ & 1111 & & \\ & & \ddots & \end{bmatrix}^{\mathrm{T}} = \mathbf{I}_5 \otimes \mathbf{1}_4 \ ,$$

along with the random vector $u_1 = (u_{1,1}, \ldots, u_{1,5})^{\mathrm{T}}$ which denotes the individual block effects. The random vector $u_2 = (u_{2,1}, \ldots, u_{2,20})^{\mathrm{T}}$ represents the twenty random block-by-treatment interaction effects and has the corresponding incidence matrix A_2 of the form $A_2 = \mathbf{I}_{20}$.

1.3.2 Nested Random Effects

Suppose data were collected from an experiment that was repeated on each of 5 consecutive days. On each day, three treatments were assigned to

three pups from each of six litters. Let Y_i be the response observed on the ith rat pup, $X_{ij} = 1$ if treatment j was applied to the ith pup and 0 otherwise (for $j = 1$, 2, and 3), $A_{1ik} = 1$ if the ith pup was tested on day k and 0 otherwise (for $k = 1, \ldots, 5$), and $A_{2il} = 1$ if the ith rat pup came from the lth litter and 0 otherwise ($l = 1, \ldots, 30$). If the mean of the ith response is μ_i, then one potential model would be

$$\begin{aligned} g(\mu_i) = \quad & \beta_1 X_{i1} + \beta_2 X_{i2} + \beta_3 X_{i3} + \\ & A_{1i1} u_{1,1} + A_{1i2} u_{1,2} + \cdots + A_{1i5} u_{1,5} + \\ & A_{2i1} u_{2,1} + A_{2i2} u_{2,2} + \cdots + A_{2i(30)} u_{2,30} \end{aligned}$$

where the $u_{1,k}$ are random day effects which are taken to be independent and identically distributed as $N(0, \sigma_1^2)$, and the $u_{2,l}$ are the random litter effects, which are taken to be independent and identically distributed as $N(0, \sigma_2^2)$, along with the mutual independence of the u_1's and u_2's.

Suppose the data are organized so that the first three observations come from treatments 1, 2, and 3 applied to the pups from litter 1, the second three come from treatments 1, 2, and 3 applied to the pups from litter 2, and so on. In this case, the design and random effects matrices would be

$$\underset{90 \times 3}{X} = \mathbf{I}_3 \otimes \mathbf{1}_{30} \ , \ \underset{90 \times 5}{A_1} = \mathbf{1}_{18} \otimes \mathbf{I}_5 \ , \text{ and } \underset{90 \times 3}{A_2} = \mathbf{1}_3 \otimes \mathbf{I}_{30} \ .$$

1.3.3 Split-Plot

Consider an experiment having 6 fields (the whole-plot experimental units) that are randomly assigned to one of 2 levels of fertilizer, and suppose each field is broken up into quadrants (i.e., sub-plot units), which are assigned randomly to one of 4 levels of insecticide. Let Y_i be the response observed on the ith quadrant, $X_{0i} = 1$, $X_{1ij} = 1$ if fertilizer j was applied to the ith quadrant and 0 otherwise ($j = 1,2$), $X_{2ik} = 1$ if insecticide k was applied to the ith quadrant and 0 otherwise ($k = 1, \ldots,4$), $A_{1il} = 1$ if the ith quadrant belongs to the lth field and 0 otherwise ($l = 1, \ldots,6$), and $A_{2ilm} = 1$ if the ith quadrant is quadrant m in the lth field and 0 otherwise ($m = 1, \ldots, 4$). If the mean of the ith response is μ_i, then one could use the model

$$
\begin{aligned}
g(\mu_i) = \quad & \beta_0 X_{0i} + \beta_1 X_{1i1} + \beta_2 X_{2i1} + \beta_3 X_{2i2} + \beta_4 X_{2i3} + \\
& \beta_5 X_{1i1} X_{2i1} + \beta_6 X_{1i1} X_{2i2} + \beta_7 X_{1i1} X_{2i3} + \\
& A_{1i1} u_{1,1} + A_{1i2} u_{1,2} + A_{1i3} u_{1,3} + \cdots + A_{1i6} u_{1,6} + \\
& A_{2i11} u_{2,1} + A_{2i12} u_{2,2} + \cdots + A_{2i14} u_{2,4} + \\
& A_{2i21} u_{2,5} + A_{2i22} u_{2,6} + \cdots + A_{2i24} u_{2,8} + \cdots A_{2i64} u_{2,24}
\end{aligned}
$$

where the $u_{1,k}$ are random field effects which are taken to be independent and identically distributed as $N(0, \sigma_1^2)$, and the $u_{2,l}$ are the random quadrant effects, which are taken to be independent and identically distributed as $N(0, \sigma_2^2)$, along with the independence of the u_1's and u_2's.

Suppose the data are organized so that the first four observations come from treatments 1, 2, 3, and 4 applied to the quadrants from field 1, the second four come from the treatments 1, 2, 3, and 4 applied to the quadrants from field 2, and so on, with fields having fertilizer 1 applied preceding those which had fertilizer 2 applied. In this case, the appropriately defined matrices would be

$$
\underset{24 \times 8}{X} = \begin{bmatrix} \mathbf{1}_{12} & \mathbf{1}_{12} & \begin{pmatrix} 1&0&0 \\ 0&1&0 \\ 0&0&1 \\ 0&0&0 \end{pmatrix} \otimes \mathbf{1}_3 & \begin{pmatrix} 1&0&0 \\ 0&1&0 \\ 0&0&1 \\ 0&0&0 \end{pmatrix} \otimes \mathbf{1}_3 \\ \mathbf{1}_{12} & \mathbf{0}_{12} & \begin{pmatrix} 1&0&0 \\ 0&1&0 \\ 0&0&1 \\ 0&0&0 \end{pmatrix} \otimes \mathbf{1}_3 & \begin{pmatrix} 0&0&0 \\ 0&0&0 \\ 0&0&0 \\ 0&0&0 \end{pmatrix} \otimes \mathbf{1}_3 \end{bmatrix}, \quad \underset{24 \times 6}{A_1} = \mathbf{I}_6 \otimes \mathbf{1}_4 \quad , \text{ and } A_2 = \mathbf{I}_{24} \quad .
$$

1.3.4 Longitudinal/Repeated Measures

Suppose one has data from a longitudinal study of 20 patients, each randomly assigned to one of 4 treatments and having measurements taken at 6 time points. Let Y_i represent the ith component of the response vector, which has mean μ_i. Let $X_{0i} = 1$, $X_{1ij} = 1$ if the ith component was from a patient given the jth treatment ($j = 1, \ldots, 4$), $X_{2ik} = 1$ if the ith components was measured at the kth time point and 0 otherwise ($k = 1, \ldots, 6$), $A_{1il} = 1$ if the ith response belongs to the lth patient and 0 otherwise ($l = 1, \ldots, 20$), and $A_{2ilm} = 1$ if the ith response was measured at time point m for the lth patient and 0 otherwise ($m = 1, \ldots, 6$). Under certain assumptions of intra-subject correlation (such as sphericity), one may consider the model

$$
\begin{aligned}
g(\mu_i) = \quad & \beta_0 X_{0i} + \beta_1 X_{1i1} + \beta_2 X_{1i2} + \beta_3 X_{1i3} + \\
& \beta_4 X_{2i1} + \beta_5 X_{2i2} + \beta_6 X_{2i3} + \beta_7 X_{2i4} + \beta_8 X_{2i5} + \\
& \beta_9 X_{1i1} X_{2i1} + \beta_{10} X_{1i1} X_{2i2} + \cdots + \beta_{23} X_{1i3} X_{2i5} + \\
& A_{1i1} u_{1,1} + A_{1i2} u_{1,2} + A_{1i3} u_{1,3} + \cdots + A_{1i(20)} u_{1,20} + \\
& A_{2i11} u_{2,1} + A_{2i12} u_{2,2} + \cdots + A_{2i16} u_{2,6} + \\
& A_{2i21} u_{2,7} + A_{2i22} u_{2,8} + \cdots + A_{2i26} u_{2,12} + \cdots A_{2i(20)6} u_{2,120}
\end{aligned}
$$

where the $u_{1,k}$ are random patient effects which are taken to be independent and identically distributed as $N(0, \sigma_1^2)$, and the $u_{2,l}$ are the random effects due to with-in patient variability, which are taken to be independent and identically distributed as $N(0, \sigma_2^2)$, along with the independence of the u_1's and u_2's.

Suppose the data are organized so that the first six observations come from time points 1, 2, ..., 6 for a patient given treatment 1, the second six come from the time points 1, 2, ..., 6 for a second patient given treatment 1, and so on with all patients given treatment 1 preceding those given treatment 2, who precede those given treatment 3 which precede those given treatment 4. The design matrix X would be a 120×24 matrix of fixed treatment, time, and treatment-by-time interaction effects. The matrices A_1 and A_2 would be

$$
\underset{120 \times 20}{A_1} = \mathbf{I}_{20} \otimes \mathbf{1}_6 \quad , \quad \text{and } A_2 = \mathbf{I}_{120} \quad .
$$

1.3.5 Sample Survey

Suppose data is available from a survey conducted across 5 geographical regions in such a way that there were 7 interviewers collecting responses in each region and 50 interviewees from each interviewer. Suppose also that covariates are available from each person interviewed (such as age, gender, socioeconomic status, etc.) and from each interviewer (age, gender, years of experience as interviewer). One may wish to analyze the response to a question using the interviewee covariates, but would like to take into account random effects due to the interviewer and region. Also, interviewer covariates may be considered fixed effects to help understand/classify interviewers. A possible model would have the random effects matrices A_1 and A_2 formed as

$$\underset{1750 \times 5}{A_1} = \mathbf{I}_5 \otimes \mathbf{1}_{350} \quad , \quad \text{and} \quad \underset{1750 \times 35}{A_2} = \mathbf{I}_{35} \otimes \mathbf{1}_{50} \quad ,$$

with u_1 representing the 5 random region effects, assumed to be independent and identically distributed as $N(0, \sigma_1^2)$, and u_2 representing the 35 random interviewer effects, assumed to be independent and identically distributed as $N(0, \sigma_2^2)$, along with the independence of the u_1's and u_2's.

1.4 Examples

The following real data sets further demonstrate the need for inferences in generalized linear models with two random effects. Numerical results are presented to illustrate that different answers may result from different estimation methods.

1.4.1 Cell Irradiation Data

This data set comes from an experiment to measure the mortality of cancer cells under radiation and was included in a paper by Robert Schall (1991). Four hundred cancer cells were placed in a dish, and three dishes were irradiated together in a radiation chamber. After the cells were irradiated, the surviving cells were counted. Since cells would die naturally, dishes with cells were put in the radiation chamber without being irradiated to establish the natural mortality. Only the 27 dishes of zero-dose exposure data were reported by Schall (1991), and are given in Table 1.

The objective of the analysis is to estimate the natural odds of survival, as well as to place a confidence interval on it, while accounting for possible random trial and dish effects. The data suggest a strong trial effect (trials 3 and 8 have rather low counts for all three respective dishes while trials 1 and 7 have high counts) and a possible dish effect (dishes do vary within a trial, but much less so than across trials — except in trial 5).

Table 1. Cell Irradiation Data

	Number of cells surviving out of 400 placed		
Trial	Dish 1	Dish 2	Dish 3
1	178	193	217
2	109	112	115
3	66	75	80
4	118	125	137
5	123	146	170
6	115	130	133
7	200	189	173
8	88	76	90
9	121	124	136

A possible approach for this data is to assume that conditional on the trial and dish effects being known, the number of surviving cells in each dish is distributed as binomial with mean $400\pi_{ij}$ ($i = 1,\ldots,9$; $j = 1,2,3$) and that

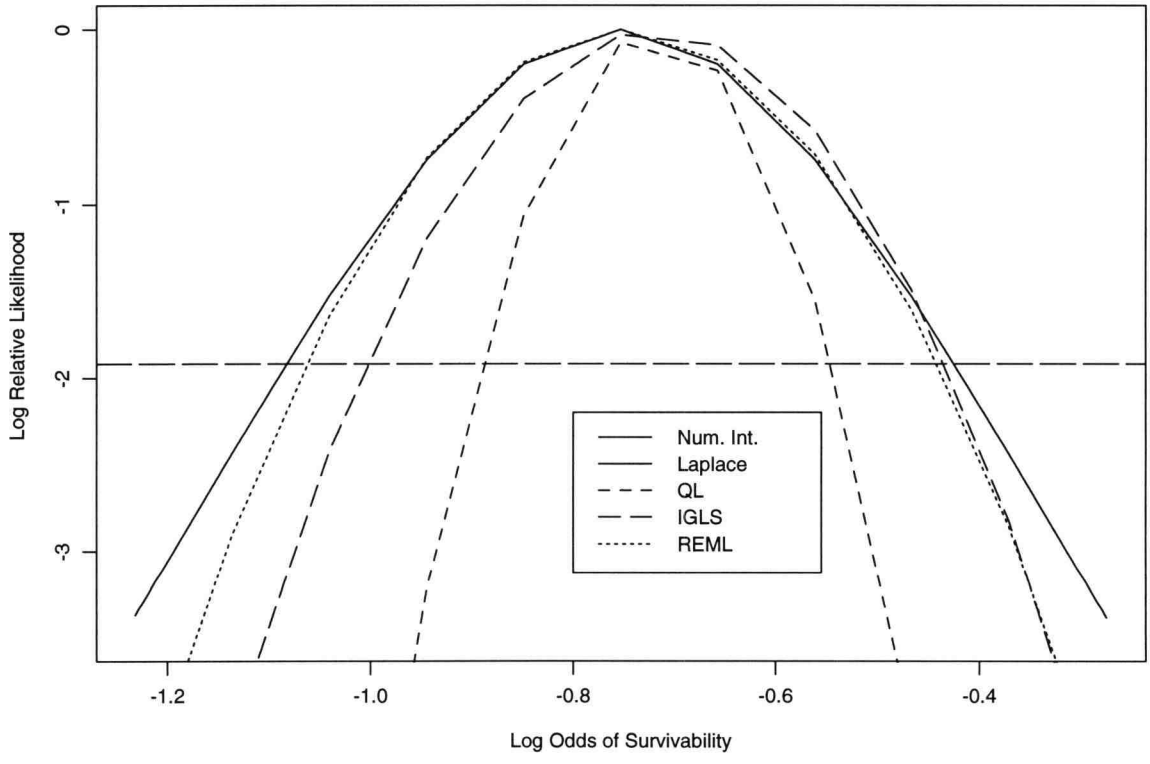
$$ln[\pi_{ij}/(1-\pi_{ij})] = \beta + u_{1i} + u_{2ij},$$

where β is the natural log-odds of survival, u_1 represents the 9 random effects due to trials and u_2 represents the 27 random effects due to each dish. It will be further assumed that $u_1 \sim N_9(\mathbf{0}, \sigma_1^2\boldsymbol{I})$, $u_2 \sim N_{27}(\mathbf{0}, \sigma_2^2\boldsymbol{I})$ and u_1 and u_2 are independent.

The maximum likelihood estimate of β, based on the marginal distribution of Y (and requiring numerical integration), is -0.7532 and a 95% *likelihood-based confidence interval* for β is -1.083 to -0.4249 (obtained by inverting a 5% level likelihood ratio test). The corresponding estimate and interval obtained by the Laplace approximation to the likelihood are identical to these.

For comparison, Figure 1 exhibits confidence intervals for β from five different methods. The solid line is the logarithm of the profile likelihood for β. The approximation based on the Laplace method is indistinguishable from this. The 95% confidence interval based on the likelihood ratio test is the set of values for which the *log profile likelihood* is greater than -1.92. Although the remaining estimation procedures— quasilikelihood (QL), iterative generalized least squares (IGLS), and restricted maximum likelihood (REML)— will be discussed in more detail in the next chapter, the confidence intervals based on them are also exhibited as the values for which corresponding curves are greater than -1.92.

Figure 1. Log Relative Profile Plot - Cell Irradiation



If the specified model is correct, then the likelihood inference is the standard to which the others should be compared. In this example it is apparent that the Laplace approximation leads to essentially exact likelihood analysis and that the other methods give different results, particularly with the lower confidence limit.

1.4.2 Acid Red 114 Revertant Colony Data

The experimental data in Table 2 was described by Simpson & Margolin (1986) and result from three replicate Ames tests in which plates containing Salmonella bacteria of strain TA98 were exposed to various doses of Acid Red 114. The number of visible revertant colonies on each plate was observed. Each replicate is from a different preparation of Hamster livers and each dose group within replicate is from a common dilution of Acid Red.

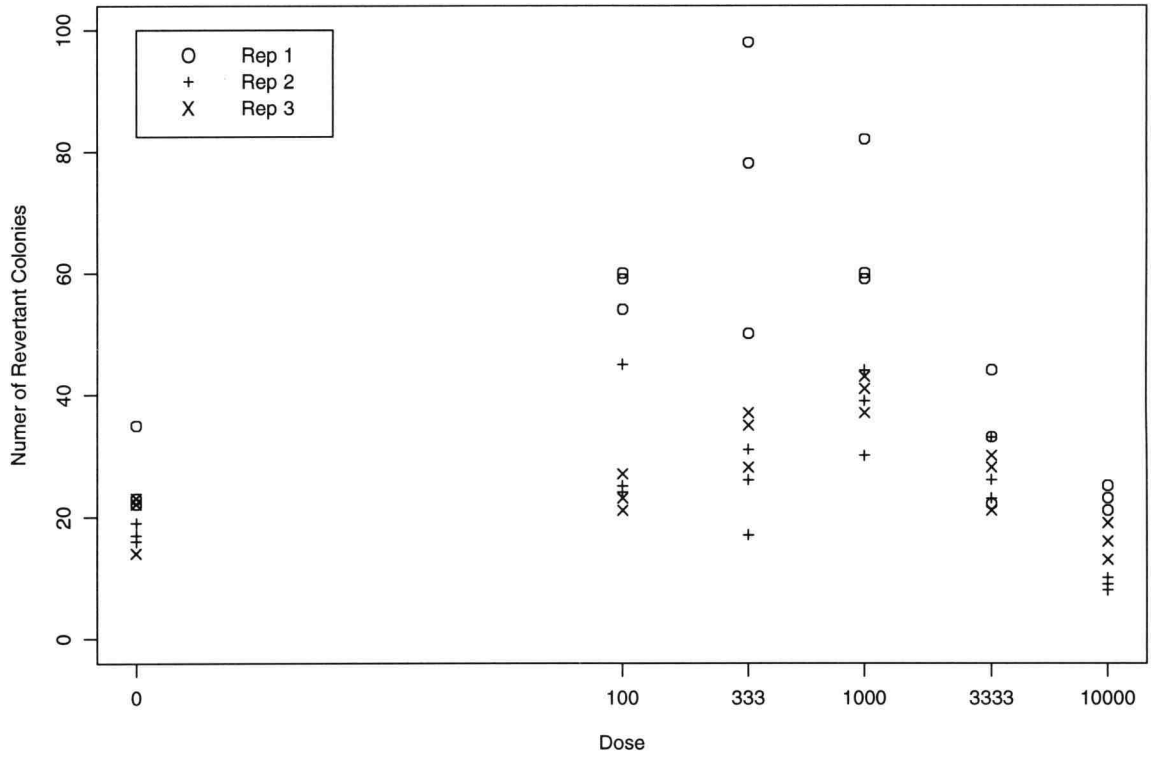
A goal of the analysis is to determine if and where the dose–response relationship begins decreasing. The data suggest an increase in response with increasing dose up to the 300–1000 μg/ml range, followed by a drop in response beyond that point (see Figure 2). Interest, here, lies in estimating the dose that maximizes the dose–response curve.

Table 2. Revertant Colony Data

	Dose (μg/ml)					
Replicate	0	100	333	1000	3333	10000
1	22	60	98	60	22	23
	23	59	78	82	44	21
	35	54	50	59	33	25
2	19	45	26	39	33	10
	17	25	17	44	26	8
	16	24	31	30	23	9†
3	23	27	28	41	28	16
	22	23	37	37	21	19
	14	21	35	43	30	13

† *In Simpson and Margolin (1986), this value was coded as missing. For easier computation, the missing value was replaced by the value 9, the average from the other two plates.*

Figure 2. Revertant Colony Data



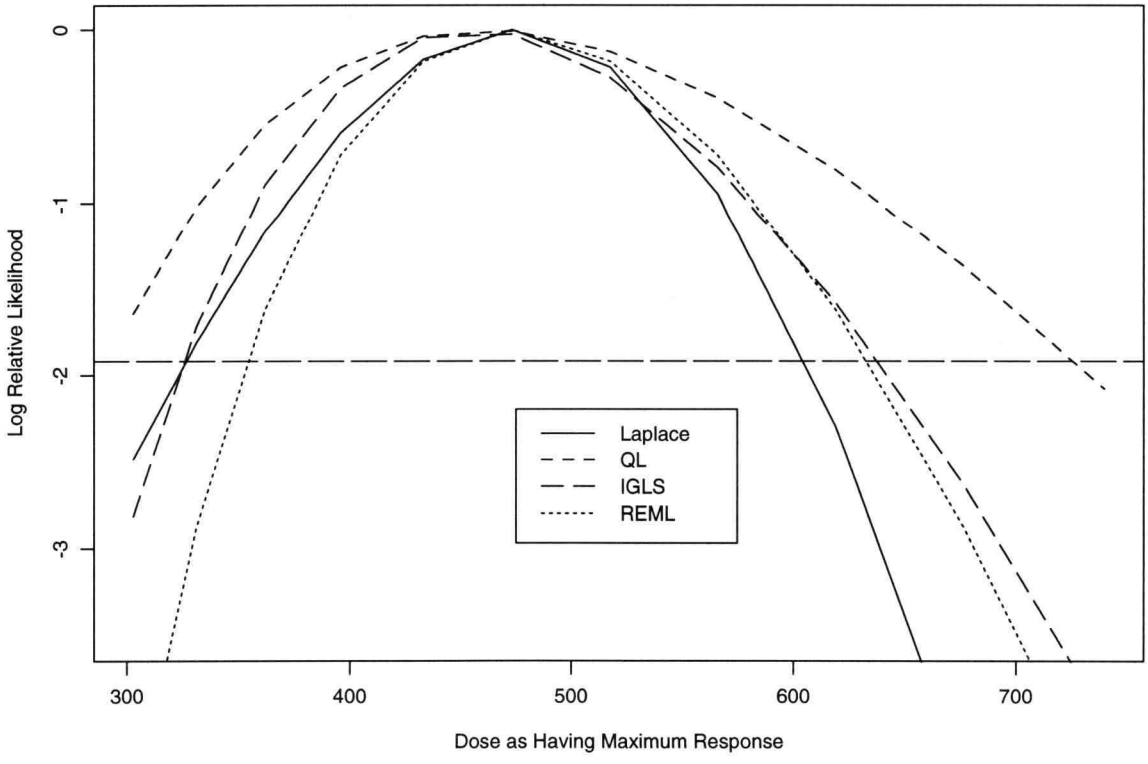
A possible model could be formulated by assuming that conditional on the replicate and plate effects being known, the number of revertant colonies on each plate is distributed as Poisson with mean μ_{ijk} ($i = 1, 2, 3$; $j = 1, 2, 3$; $k = 1,\ldots,6$) and that

$$ln(\mu_{ijk}) = \beta_0 + \beta_1 D_{ijk} + \beta_2 D_{ijk}^2 + \beta_3 Z_{ijk} + u_{1i} + u_{2jk(i)}$$

where $D = ln(\text{Dose} + 1)$, $Z = \begin{cases} 1 & \text{if Dose } = 0 \\ 0 & \text{otherwise} \end{cases}$, u_1 represents the 3 random effects due to replicate and u_2 represents the 54 random effects due to each plate. It will be further assumed that there is no dose–by–replicate interaction, that $u_1 \sim N_3(\mathbf{0}, \sigma_1^2 \boldsymbol{I})$, $u_2 \sim N_{54}(\mathbf{0}, \sigma_2^2 \boldsymbol{I})$, and that u_1 and u_2 are independent.
It is desired to carry out inferences about β_1 and β_2 in this model. For example, the value of D that maximizes the response is $-\beta_1/2\beta_2$. It is desired to test $\beta_2 = 0$ (i.e., there is no downturn in the dose-response curve) and, if rejected, to estimate the dose, along with confidence limits, at which the downturn occurs.

This data set was analyzed using approximate maximum likelihood based on a Laplace approximate likelihood, Wedderburn's (1974) quasi-likelihood (QL) model, an iterative generalized least squares (IGLS) routine based on approximate moments, and Schall's (1991) REML method. The use of numerical integration to obtain exact maximum likelihood estimates would be prohibitively time consuming due to the dimensionality of the required integrals (i.e., on $\mathbb{R}^{19}$). The moment models (QL, IGLS, REML) and maximum likelihood method (Laplace) gave very similar estimates for all 4 β parameters and differed slightly in dispersion component estimation (see Table 15 in Chapter 4). The maximum likelihood estimate (473.42 μg/ml) and 95% *likelihood-based confidence interval* (326.67 to 603.62 μg/ml) for the dose yielding the maximum response are based on maximizing the Laplace approximate likelihood. Plotted in Figure 3 is the log relative profile likelihood for $-\beta_1/2\beta_2$ (rescaled to Dose) along with the pseudo profile likelihoods corresponding to the QL, IGLS, and REML models. As in Figure 1, it is apparent that the methods give different results and the simplistic quasilikelihood approach can give quite differing conclusions.

Figure 3. Log Relative Profile Plot - Revertant Colony



1.4.3 Salamander Mating Data

A data set was included in the book by McCullagh and Nelder (1989, Ch. 14), in which male and female salamanders of two populations from different geographic locations were paired in an incomplete Latin-square design to see if barriers developed over time to inhibit successful mating across populations. Females from two distinct populations called whiteside (W) and rough-butt (R) were paired for mating with males from their own population and from the other population. A primary interest was to estimate the probability of a successful mating for each of the four combinations of gender and population. Only the first of three such experiments reported in McCullagh and Nelder will be used here. The experimental setup is laid out in Table 3, while the data are given in Table 4.

The first row of Table 3 states that rough-butt Female 1 was matched with rough-butt Males 1, 5 and 4 on June 4, June 12 and June 20; and with white-side Males 4, 1 and 5 on June 8, June 16 and June 24, respectively. Correspondingly, the sixth row shows that rough-butt Female 6 was matched with rough-butt Males 9, 7 and 8 on June 8, June 16 and June 24; and with white-side Males 9, 10 and 6 on June 4, June 12 and June 20. Similarly, white-side Female 1 was matched with rough-butt Males 9, 7 and 10 on June 4, June 12 and June 20; and with white-side Males 9, 10 and 8 on June 8, June 16 and June 24. White-side Female 6 was matched with rough-butt Males 2, 4 and 1 on June 8, June 16 and June 24; and with white-side Males 5, 3 and 2 on June 6, June 12 and June 20. Thus, each female and male salamander is involved in only 6 of the possible 20 matches.

From the data in Table 4, rough-butt Female 1 was involved in 5 successful matings in the 6 matches placed in, with a failed mating with white-side Male 1 (on June 16). Rough-butt Female 6 also had 5 successful matings; the failure occurring with rough-butt Male 7 (on June 16). White-side Female 1 had 4 successful matings, and failures with rough-butt Males 9 and 10; while white-side Female 6 had only a single successful mating, and that was with white-side Male 3. (Some tabular summaries of the data can be found in Tables 14.7 and 14.8 in McCullagh and Nelder, 1989.)

Table 3. Salamander Mating: Design Layout

		(Entries are male identification codes)					
Females		June 4	June 8	June 12	June 16	June 20	June 24
	1	1	4	5	1	4	5
	2	5	5	3	3	1	2
R	3	R 2	W 1	R 1	W 4	R 3	W 3
	4	4	2	2	5	5	4
	5	3	3	4	2	2	1
	6	9	9	10	7	6	8
	7	8	8	9	9	7	6
R	8	W 6	R 6	W 7	R 10	W10	R 9
	9	10	7	8	6	8	10
	10	7	10	6	8	8	7
	1	9	9	7	10	10	8
	2	7	6	9	7	6	10
W	3	R 8	W 7	R 6	W 9	R 7	W 6
	4	10	10	8	8	9	9
	5	6	8	10	6	8	7
	6	5	2	3	4	2	1
	7	4	1	5	2	1	5
W	8	W 1	R 4	W 2	R 5	W 5	R 3
	9	3	3	1	1	4	4
	10	2	5	4	3	3	2

The objective of the following analysis is to estimate the odds of successful mating for the 4 possible male/female–rough-butt/white-side matchings, with confidence limits, while taking into account possible random individual male and female salamander effects. A possible approach for modeling the experimental results is to assume that conditional on the male and female effects being known, the mating failure/success outcome between Female i and Male j is a Bernoulli random variable with probability of success denoted by π_{ij} ($i = 1,\ldots,20$; $j = 1,\ldots,20$) and that

$$ln[\pi_{ij}/(1-\pi_{ij})] = \beta_0 + \beta_1 F_i + \beta_2 M_j + \beta_3 FM_{ij} + u_{Fi} + u_{Mj} \quad ,$$

Table 4. Salamander Mating: Observed Matings

(Entries of 1 represents a successful mating)

Females		June 4	June 8	June 12	June 16	June 20	June 24
	1	1	1	1	0	1	1
	2	1	1	1	1	1	1
R	3	R 1	W 0	R 1	W 1	R 1	W 1
	4	1	1	1	0	1	1
	5	1	1	1	1	1	1
	6	1	1	1	0	1	1
	7	0	0	0	1	0	0
R	8	W 0	R 1	W 0	R 0	W 1	R 1
	9	0	0	1	1	1	1
	10	0	0	1	0	1	0
	1	0	1	1	1	0	1
	2	0	0	0	1	0	0
W	3	R 0	W 0	R 0	W 0	R 0	W 1
	4	0	1	1	1	0	1
	5	0	1	0	0	0	0
	6	0	0	1	0	0	0
	7	1	1	1	0	1	1
W	8	W 1	R 0	W 1	R 0	W 1	R 0
	9	1	1	1	1	1	0
	10	1	0	0	1	1	0

where $F_i = \begin{cases} 1 & \text{if Female } i \text{ is white-side} \\ 0 & \text{otherwise} \end{cases}$,

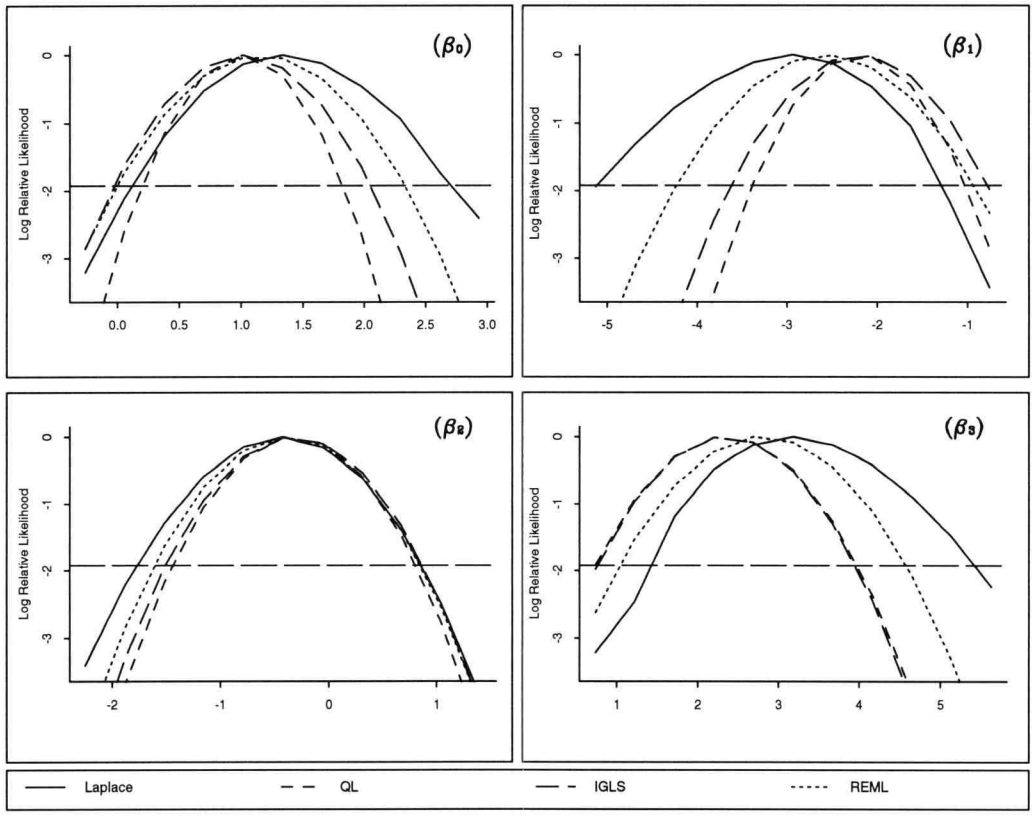
$M_j = \begin{cases} 1 & \text{if Male } j \text{ is white-side} \\ 0 & \text{otherwise} \end{cases}$,

$FM_{ij} = \begin{cases} 1 & \text{if Female } i \text{ and Male } j \text{ are white-side} \\ 0 & \text{otherwise} \end{cases}$,

u_{Fi} represents the 20 female random effects and u_{Mj} represents the 20 male random effects. It will be further assumed that $u_F \sim N_{20}(\mathbf{0}, \sigma_F^2 \boldsymbol{I})$, $u_M \sim N_{20}(\mathbf{0}, \sigma_M^2 \boldsymbol{I})$ and u_F and u_M are independent. Implicit in the above is that there are no population nor population-by-sex random effects.

It is desired to estimate and draw inferences regarding the four β parameters in the model. The data set was analyzed using QL, IGLS, REML, and approximate maximum likelihood based on the Laplace approximate likelihood. Due to the structure of the random effects, maximum likelihood via numerical integration is impractical, requiring numerical integration over $\mathbb{R}^{20}$. Figure 4 presents the log relative profile curves for all 4 fixed effects, based on the Laplace approximate log likelihood, along with pseudo profile likelihoods corresponding to QL, IGLS, and REML. (Estimates and confidence interval endpoints can be found in Table 16 of Chapter 4.) Apart from β_2, there is some disparity among the methods' profile plots. In particular, confidence intervals for the interaction term β_3 are quite different. These differences will be discussed in more detail in Chapter 4.

Figure 4. Log Relative Profile Plots - Salamander Mating Experiment 1



1.5 Overview of the Rest of the Thesis

The examples given in Section 1.4 demonstrate that differing results come from the different methods. From the first example, it appears that maximum likelihood using the Laplace approximation to the likelihood function yields exact results. It is also apparent that REML can give nearly the same results. This thesis investigates the methods described in the examples. Chapter 2 discusses why Model (1.1) given in Section 1.2 is worthy of research, as well as presents developments of approaches to addressing the problem. There, IGLS and REML are formally developed. In Chapter 3, the Laplace approximation to the likelihood function is laid out, as well as its use for estimation and inferences. Some comparisons with the Laplace approximate likelihood analysis and exact maximum likelihood via numerical integration are also made. Chapter 4 presents analysis results in more depth for the examples introduced in Section 1.4, as well as summarizes results of some simulation analyses. Chapter 5 presents conclusions from the research and suggestions for further research. Computer programs for estimation and inference are given in the Appendices.

2. RANDOM EFFECTS IN GENERALIZED LINEAR MODELS

2.1 Generalized Linear Models

Generalized Linear Models (GLMs) extend linear modeling techniques to data that are non-normal but whose distribution is a member of the regular exponential family. The regression models may be nonlinear, but the non-linearity is contained in a "link" function of the mean and regression parameters (coefficients). Since the normal distribution is a member of this class, ordinary linear models are a special case of GLMs. The most commonly used of the other GLMs are Binomial logistic, Binomial probit, and Poisson log-linear regression.

2.1.1 Model Definition and Parameter Estimation

Let Y represent an n by 1 vector of responses and X a known n by k matrix of explanatory variables. A GLM is specified by:

1. The elements of Y are independent of one another with probability density functions:

$$f(y_i;\, \theta_i,\, \phi) = exp\{[y_i\theta_i - b(\theta_i)]/(\phi a_i) + c(y_i, a_i)\}$$

for specified functions $b(\ )$ and $c(\ )$, and with a_i's being known constants. It follows that

$$\mathrm{E}\{y_i\} = \mu_i = b'(\theta_i) \text{ and } \mathrm{Var}\{y_i\} = \phi a_i V(\mu_i)\ ,$$

where $V(\mu_i) = b''(\theta_i)$, and $\mathrm{b}'(\ )$ and $\mathrm{b}''(\ )$ denote first and second derivatives. If ϕ is known this is a one-parameter exponential family distribution with canonical parameter θ.

2. The mean is related to the explanatory variables through the link function, $g(\ )$, by $g(\mu_i) = x_i^{\mathrm{T}}\beta$, where β is a k by 1 vector of coefficients and $^{\mathrm{T}}$ denotes matrix transposition.

Because of the prominent role of iteratively reweighted least squares (IWLS) methods in this thesis, it is useful to examine the nonlinear least squares solution to the mean and variance model implied by 1 and 2 above. That this produces the same estimates as maximum likelihood is well known (see, for example, McCullagh and Nelder 1989, Section 2.5). Since the model may be written as

$$y_i = g^{-1}(x_i^{\mathrm{T}}\beta) + \epsilon_i \ , \tag{2.1}$$

where the ϵ_i's are independent random errors with $\mathrm{E}\{\epsilon_i\} = 0$ and $\mathrm{Var}\{\epsilon_i\} = \phi a_i V(\mu_i)$, the Gauss-Newton method involves the iterative fitting, by weighted least squares, of the linearized model. The basic idea is: (1) linearize the mean function in terms of the parameter vector β; (2) move "constant" terms to the left hand side, forming a new dependent variate; (3) regress this new variate on the resulting linear model; and (4) repeat steps 2 and 3 until convergence. Using $h(\ )$ to represent $g^{-1}(\ )$, this linearization about $\beta = \beta^{(t)}$ (where $\beta^{(t)}$ is a current estimate of β after t iterations) is:

$$y_i \simeq h\Big(x_i^{\mathrm{T}}\beta^{(t)}\Big) + h'\Big(x_i^{\mathrm{T}}\beta^{(t)}\Big)x_i^{\mathrm{T}}[\beta - \beta^{(t)}] + \epsilon_i \ ,$$

or, equivalently,

$$z_i^{(t)} \simeq x_i^{\mathrm{T}}\beta + \epsilon_i^{(t)} \tag{2.2}$$

where

$$z_i^{(t)} = \frac{y_i - h\Big(x_i^{\mathrm{T}}\beta^{(t)}\Big)}{h'\Big(x_i^{\mathrm{T}}\beta^{(t)}\Big)} + x_i^{\mathrm{T}}\beta^{(t)} \ , \ \epsilon_i^{(t)} = \frac{\epsilon_i}{h'\Big(x_i^{\mathrm{T}}\beta^{(t)}\Big)} \ ,$$

$$\mathrm{E}\{\epsilon_i^{(t)}\} = 0 \ , \ \mathrm{Var}\{\epsilon_i^{(t)}\} \simeq \frac{\phi a_i V(\mu_i^{(t)})}{\Big[h'\Big(x_i^{\mathrm{T}}\beta^{(t)}\Big)\Big]^2} \ , \text{ and } \mu_i^{(t)} = h\Big(x_i^{\mathrm{T}}\beta^{(t)}\Big) \ .$$

Thus weighted least squares, with weights being the inverse of the estimates of $\mathrm{Var}\{\epsilon_i^{(t)}\}$, is used to update $\beta^{(t)}$ to $\beta^{(t+1)}$. An initial estimate of β is unnecessary with $z_i^{(0)} = \mu_i^{(0)} = g(y_i)$, leading to $h'\Big(x_i^{\mathrm{T}}\beta^{(0)}\Big) = 1/g'(y_i)$ and $V(\mu_i^{(0)}) = V(g(y_i))$.

Thus the following IWLS routine is suggested for estimating β, with ϕ known:

1] With $\beta^{(t)}$ compute the working "response" vector, $\tilde{Z}$, and diagonal weight matrix, $\tilde{W}$, via

$$\tilde{Z} = X\beta^{(t)} + H^{-1}\left[y - h\left(X\beta^{(t)}\right)\right]$$

$$\tilde{W}_{ii} = \frac{\phi a_i V\left(h\left(x_i^{\mathrm{T}}\beta^{(t)}\right)\right)}{\left[h'\left(x_i^{\mathrm{T}}\beta^{(t)}\right)\right]^2} \quad ,$$

where $H = diag\left[h'\left(x_i^{\mathrm{T}}\beta^{(t)}\right)\right]$.

2] Compute new estimate of β, $\beta^{(t+1)}$, via

$$\beta^{(t+1)} = (X^{\mathrm{T}}\tilde{W}^{-1}X)^{-1}X^{\mathrm{T}}\tilde{W}^{-1}\tilde{Z} \quad .$$

3] Go to [1] using $\beta^{(t+1)}$ as the current estimate of β until convergence.

In the scheme above, step [2] obtains a weighted least squares estimate for β using estimated weights. Since these weights are functions of β, one must iterate with new working responses and weights. Notice that ϕ could have been assumed to be 1 with no affect on $\hat{\beta}$, the final estimate of β. This is due to ϕ being scaler, which passes through inversion of matrices, and cancels itself out in step [2]. In the case when ϕ is not known, $\tilde{W}_{ii}$ is often computed assuming $\phi = 1$ and the IWLS scheme above is carried out with no further changes. Once β has been estimated, (with $\hat{\beta}$), ϕ can be estimated by the following method of moments (MOM) estimator (McCullagh and Nelder, 1989 p. 328):

$$\hat{\phi} = \frac{1}{n-k}\sum_{i=1}^{n}\frac{[y_i - h(x_i^{\mathrm{T}}\hat{\beta})]^2}{a_i V(h(x_i^{\mathrm{T}}\hat{\beta}))} \quad ,$$

where k is the number of elements in β.

2.1.2 Difficulty in Incorporating Random Effects

Unlike normal linear models, in which the response may be written as the sum of fixed terms and normally-distributed random terms, GLMs allow for the specification of randomness only through the distribution of the response variable. The addition of random components to represent, for example, block effects in a randomized block experiment, requires a departure from the standard GLM definition. Also, there are choices involved in how the random effect should be incorporated. For example, the random effects may be additive to the mean or additive on the same scale as the fixed effects.

Consider an experiment in which litters of K mice are randomized to one of T treatments, with B litters per treatment. Suppose that a normally-distributed response is measured on each mouse. This corresponds to a linear mixed model with a single fixed effect and nested random effects:

$$y_{ijk} \sim N(\mu_{ij}, \sigma^2) \ , \ \mu_{ij} = \tau_i + u_{ij} \ , \ (i = 1,\ldots,T;\ j = 1,\ldots,B;\ \ k = 1,\ldots,K) \ ,$$

where τ_i represents the effect of treatment level i and u_{ij} represents the effect of litter j in treatment level i. This model is more commonly written as

$$y_{ijk} = \tau_i + u_{ij} + e_{ijk} \quad ,$$

with the assumptions $u_{ij} \sim N(0,\ \sigma_u^2)$, $e_{ijk} \sim N(0,\ \sigma^2)$ and the u's are mutually uncorrelated with the e's. This model allows the mean to vary across treatments and for correlation among mice within litters.

The marginal distribution of the y's is easily shown to be normal with

$$\mathrm{E}\{y_{ijk}\} = \tau_i \quad \text{and } \mathrm{Cov}\{y_{ijk},\ y_{lmn}\} = \begin{cases} 0 & (i,j) \neq (l,m) \\ \sigma_u^2 & (i,j) = (l,m), k \neq n \\ \sigma_u^2 + \sigma^2 & (i,j,k) = (l,m,n) \end{cases} .$$

Maximum likelihood estimates and restricted maximum likelihood estimates for the τ's, σ_u^2 and σ^2 can be obtained by generalized least squares (Harville, 1977; Schall, 1991).

Now suppose instead that a binary response is measured on each mouse. There are several choices for models which consider the litter effects as random. A natural way to model the litter effects is to incorporate them as additive to the fixed effects (e.g., $g(\pi_{ij}) = \tau_i + u_{ij}$, where $\pi_{ij} = \mathrm{E}\{y_{ijk} | \, u_{ij}\}$). Pierce and Sands (1975) introduced this model for a random litter effects problem, assuming as well that $u_{ij} \sim N(0, \sigma_u^2)$. The model has been further developed for other situations (Stiratelli, Laird and Ware, 1984; Wong and Mason, 1985). A drawback to this modeling approach is that the likelihood function for τ_i and σ_u^2 is intractable (see Section 2.2.1), and therefore can be computationally cumbersome, even for moderately sized data sets. Still, this model has some very appealing aspects, the foremost of which is its ease in extending to several random effects (see Section 2.3). This model has been labeled logistic-normal or probit-normal, depending on whether the logit or probit link is used (Searle, Casella and McCulloch, 1992).

Alternatively the beta-binomial model, which yields a tractable likelihood function for τ_i, has also been suggested for the random litter effects problem. Here a parametric distribution for π_{ij} is assumed, and the treatment effects are modeled into the moments of π_{ij}. More specifically, assume

$$y_{ijk} \, | \, \pi_{ij} \sim \text{Bernoulli}(\pi_{ij}) \quad , \quad g(\mathrm{E}\{\pi_{ij}\}) = \tau_i$$

and

$$\pi_{ij} \sim \text{beta}(\alpha_i, \gamma_i) \quad .$$

The resulting marginal distribution of the y's is beta-binomial, from which maximum likelihood estimates for the τ's may be found (Searle, Casella and McCulloch, 1992 Section 10.3). Estimates may also be obtained from IWLS using the marginal moments (Williams, 1982).

The beta-binomial model above induces only positive intra-litter correlations. A similar approach, correlated Bernoulli trials, yields the same marginal moments yet allows negative correlations. The correlated Bernoulli model (Prentice, 1988; Haseman and Kupper, 1979) assumes that mice in the same litter tend to be correlated while the beta-binomial assumes them to be independent conditional on the random litter-specific mean. Neither model, however, easily extends to multiple random effects since the variation in the conditional mean is modeled with a single distribution.

There does not appear to be any good way to distinguish between the logistic-normal, beta binomial and correlated Bernoulli trials models based on the data alone. Haseman and Kupper (1979) compare the beta-binomial and correlated binomial, finding very little differences between their respective fits to data. Williams (1982, Section 5) says that each appears to fit equally well to data having proportions in the range of .2–.8; while for more extreme proportions, some differences in fit may be found between the logistic-normal model and and the other two. In this thesis, an important issue is that the logistic-normal is the model that readily includes additional random terms.

2.2 Estimation for Generalized Linear Models with a Single Random Effect Added to the Fixed Effects

In a technical report by Pierce and Sands (1975) the logistic-normal model that included a random effect additively in the link was introduced. Their argument was based on its simplicity in modeling and that, on some transformation, random effects should affect the mean response additively. Thus they chose the same scale as for the fixed effects. Also, if the random effects were measured, they would typically be included as an offset in the linear predictor with the fixed effects. For estimation and testing, Pierce and Sands suggest using approximate marginal moments, thus avoiding the required numerical integration for obtaining maximum likelihood estimates (MLEs). The idea of modeling all effects in the link has been applied to Poisson data (Breslow, 1984) and longitudinal/repeated measures studies with binary responses (Stiratelli, Laird and Ware, 1984).

2.2.1 Model Definition

Much attention will be focused on the model which includes a random effect on the same scale as the fixed effect. Let Y represent an n by 1 vector of responses, X a known n by k matrix of explanatory variables, and u an n by 1 vector of unknown random components. Suppose that conditionally on u_i, y_i follows a one-parameter exponential family distribution:

$$f(y_i|\, u_i;\, \theta_i) = exp\{[y_i\theta_i - b(\theta_i)]/a_i + c(y_i, a_i)\} \ .$$

Suppose also that the mean is related to the explanatory variables and random effects through the link function, $g(\ )$, by $g(\mu_i) = x_i^{\mathrm{T}}\beta + u_i$, where β is a k by 1 vector of coefficients. Assume as well the u's are independently distributed as $N(0,\ \sigma_u^2)$.

The marginal distribution of y_i can be found by integrating out the random effects in their joint distribution with the y's. With a normality assumption made about the random effects, the marginal distribution can be written as (up to a known multiplicative constant):

$$f(Y;\beta,\sigma_u^2) = \prod_i f(y_i; x_i^{\mathrm{T}}\beta, \sigma_u^2)$$

$$= \prod_i \left[(2\pi\sigma_u^2)^{\frac{-1}{2}} \int_{-\infty}^{\infty} exp\left\{ \frac{y_i\theta_i - b(\theta_i)}{a_i} - \frac{u_i^2}{2\sigma_u^2} \right\} \mathrm{d}u_i \right] \qquad (2.3)$$

where θ_i is a known function of β and u_i.

This likelihood function (2.3) for β and σ_u^2 cannot be evaluated analytically due to the nonlinearity of u_i in θ_i and $b(\theta_i)$. If $y_i|u_i$ is assumed normal, then θ_i and $b(\theta_i)$ are both linear in u_i, and the integrand can be replaced by a normal density and easily evaluated. Some discussion of (2.3) regarding maximum likelihood estimation, moment approximations, and quasi-likelihood estimation will be given in the following sub-sections as a foundation for the approaches presented in Section 2.3 for the two random effects model.

2.2.2 Maximum Likelihood Estimation

Numerical integration of (2.3) can be accomplished by Gauss-Hermite quadrature. Re-write each integral as

$$\int F(u_i) exp\left\{ -\frac{u_i^2}{2\sigma_u^2} \right\} \mathrm{d}u_i \quad ,$$

where $F(u_i) = exp[(y_i\theta_i - b(\theta_i))/a_i]$, the conditional density of y_i considered now as a function of u_i. Gauss-Hermite quadrature approximates the integral by a sum of weighted functional evaluations of F:

$$\int F(u_i) exp\left\{ -\frac{u_i^2}{2\sigma_u^2} \right\} \mathrm{d}u_i \simeq \sqrt{2\sigma_u^2} \sum_{l=1}^{m} w_l F(\sqrt{2\sigma_u^2}\, v_l) \quad ,$$

where the weights, w_l, and points of evaluation (or nodes), v_l, come from identities of the Hermitian polynomials (see Abramowitz and Stegun, 1972 p. 924). This is exact if $F(\ )$ is a polynomial in u_i of degree up to $2m-1$; i.e., if 2 nodal points are used and F is a polynomial of degree 3 or less, the approximation will be exact. Thus, the more nodal points used, the better the approximation. The drawback is that the more nodes used, the more computing required. For a general multi-dimensional integral over, say, r dimensions, using m nodes would require m^r functional evaluations of F to compute the likelihood function at each choice of β and σ_u^2. Application of Gauss-Hermite quadrature to each of the n integrals in (2.3), however, requires only m evaluations at each β and σ_u^2, as they may be carried out simultaneously due to the marginal independence of the y's. To obtain MLEs, a grid search is employed, evaluating the integral at different values for β and σ_u^2, until a maximum is found. (See Section 3.2.2 for further discussion.)

The EM algorithm (Dempster, Laird, and Rubin, 1977) has also been used for obtaining MLEs. Anderson and Aitkin (1985) suggest its use in a logistic-normal model for survey data, considering interviewer effects as random. For this method the joint density of Y and u, or $ln\ f(y,\ u)$, is optimized (the M step) in obtaining MLEs of β and σ_u^2. But since the u's are unknown (or missing) the part of the log likelihood depending on u is replaced by its expected value (obtained in the E step) given the observed y's and the current estimates of β and σ_u^2. The difficulty arises here in the E step where computation of $E\{ln\, f(y,u)|\, y\}$ is required. This is seen by noting

$$E\{ln\ f(y,u)|\, y\} = E\{ln\ f(y|\, u) + ln\ f(u)|\, y\}$$

$$= \int [ln\ f(y|\, u) + ln\ f(u)] f(u|\, y)\mathrm{d}u = \int [ln\ f(y|\, u) + ln\ f(u)] \frac{f(y|\, u) f(u)}{f(y)}\mathrm{d}u$$

$$= y^{\mathrm{T}} X\beta - \tfrac{1}{2} ln|D| + \frac{\int s(u) f(y|\, u) exp\Big\{ - u^{\mathrm{T}} u/2\sigma_u^2 \Big\}\mathrm{d}u}{\int f(y|\, u) exp\Big\{ - u^{\mathrm{T}} u/2\sigma_u^2 \Big\}\mathrm{d}u}$$

where

$$s(u) = y^{\mathrm{T}} Au - 1^{\mathrm{T}} b(X\beta + Au) - \frac{u^{\mathrm{T}} u}{2\sigma_u^2}\ .$$

The resulting numerator and denominator integrals are intractable, for which Anderson and Aitkin (1985) suggest using Gauss-Hermite quadrature for

numerical integration of each. Although these integrals are written here as multidimensional, they can be converted to the sum of univariate integrals due to the independence of data and random effects. The EM method, if it converges, yields MLEs; yet the likelihood is not found as a byproduct, limiting its use for inferences. The reason for using the EM algorithm is that the M step is rather easy to carry out (see Anderson and Aitkin, 1985 Section 3) and should converge with fewer steps than a grid search, hopefully reducing the computing time. The drawbacks, however, are the computations needed in the E step (which, by the way, includes those required for the direct MLE computations discussed above) and that inferences are not a simple byproduct.

Recently, the use of the Laplace Approximation (Barndorf-Nielson and Cox, 1989 Chapter 3; Thisted, 1988 Chapter 5) to integrals of the form (2.3) has received much attention (Breslow and Clayton, 1993; Liu and Pierce, 1993). The Laplace method approximates (2.3) by

$$f(Y;\,\beta,\,\sigma_u^2) \simeq (2\pi\sigma_u^2)^{\frac{-n}{2}}\Big[\prod_i f(y_i|\tilde{u}_i)exp\Big\{-\tilde{u}_i^2/2\sigma_u^2\Big\}\Big]\prod_i\Big[\sigma_u^{-2}+\frac{f'(y_i|\tilde{u}_i)^2}{f(y_i|\tilde{u}_i)^2}-\frac{f''(y_i|\tilde{u}_i)}{f(y_i|\tilde{u}_i)}\Big],$$

where f' and f'' denote differentiation with respect to u_i, and $\tilde{u}_i$ is the solution to

$$\frac{f'(y_i|\tilde{u}_i)}{f(y_i|\tilde{u}_i)}-\frac{\tilde{u}_i}{\sigma_u^2}=0 \quad .$$

(See Chapter 3 for more details.)

Liu and Pierce (1993) have demonstrated numerically, for the single random effects model, that the relative error in the Laplace approximation is nearly constant over the parameter space. Thus, maximum likelihood estimates and likelihood ratio inferences based on the approximation are essentially exact. Breslow and Clayton (1993) proposed a further approximation to simplify the maximization process. Their resulting estimating equations were the same as restricted maximum likelihood (see Section 2.3.3), as well as empirical Bayes of Stiratelli, Laird and Ware (1984) for longitudinal binary data.

It should be noted, however, that the ratio of integrals in the E step of the EM algorithm is exactly the form for which Tierney and Kadane (1986) used

Laplace's method and found the errors to cancel out from the numerator and denominator. Thus, using Laplace rather than numerical integration may alleviate computing time in the E step of the EM algorithm, and may be useful for drawing inferences with a final approximation of (2.3). This has not been looked into nor will it be addressed further here.

2.2.3 Estimation Based on Moment Assumptions

Alternatives to maximum likelihood estimation based on (2.3) have been considered since maximum likelihood estimation is cumbersome and depends on distributional assumptions. Two such approaches are considered in this thesis, one below and another presented in Section 2.2.4. The first is based on the approximate 1^{st} and 2^{nd} moments of the marginal distribution of Y in model (2.3). This was suggested by Pierce and Sands (1975), and later by Williams (1982) and Breslow (1984). The basic idea is analogous to the Gauss-Newton implementation described in Section 2.1.1. First, linearize the conditional mean function in the random effects. Next, obtain the approximate marginal mean and variance for the data, Y. Finally, apply the Gauss-Newton method on the resulting approximate marginal moments model.

We may write the response variable y_i as the sum of its mean, which depends on u_i, and random deviation about the mean:

$$y_i = h(x_i^{\mathrm{T}}\beta + u_i) + \epsilon_i \quad ,$$

where $h(\ ) = g^{-1}(\ )$. By expanding $h(x_i^{\mathrm{T}}\beta + u_i)$ about $u_i = 0$ we have the approximation, for small σ_u^2:

$$y_i \simeq h(x_i^{\mathrm{T}}\beta) + h'(x_i^{\mathrm{T}}\beta)u_i + \epsilon_i^* \quad ,$$

where ϵ_i^*'s are independent random errors with $\mathrm{E}\{\epsilon_i^*\} = 0$ and $\mathrm{Var}\{\epsilon_i^*\} = a_i V(\mu_i^*)$, $\mu_i^* = h(x_i^{\mathrm{T}}\beta)$. From this linearization, the approximate marginal mean and variance of y_i are easily found to be

$$\mathrm{E}\{y_i\} \simeq \mu_i^* = h(x_i^{\mathrm{T}}\beta) \ \text{ and } \mathrm{Var}\{y_i\} \simeq a_i V(\mu_i^*) + \sigma_u^2\left[h'(x_i^{\mathrm{T}}\beta)\right]^2 . \qquad (2.4)$$

If σ_u^2 were known, an iterative weighted least squares scheme like that in Section 2.1.1 could be used. At the t^{th} iteration update the estimate of β by weighted least squares regression of $z_i^{(t)}$ on x_i:

$$z_i^{(t)} = x_i^{\mathrm{T}}\beta + \epsilon_i^{**}$$

where

$$z_i^{(t)} = \frac{y_i - h\left(x_i^{\mathrm{T}}\beta^{(t)}\right)}{h'\left(x_i^{\mathrm{T}}\beta^{(t)}\right)} + x_i^{\mathrm{T}}\beta^{(t)}, \quad \mathrm{E}\{\epsilon_i^{**}\} = 0 \quad ,$$

and

$$\mathrm{Var}\{\epsilon_i^{**}\} = \sigma_u^2 + \frac{a_i V(\mu_i^{*(t)})}{\left[h'\left(x_i^{\mathrm{T}}\beta^{(t)}\right)\right]^2} \quad .$$

Since σ_u^2 is often unknown, it is updated at each iteration by a method of moments estimator. One choice is (McCullagh & Nelder, 1989 Ch. 14)

$$\sigma_u^{2(t)} = \frac{\sum_i [y_i - h(x_i^{\mathrm{T}}\beta^{(t)})]^2 - \sum_i a_i V(\mu_i^{*(t)})}{\sum_i [h'(x_i^{\mathrm{T}}\beta^{(t)})]^2} \quad .$$

This gives the following IWLS scheme for estimating β and σ_u^2:

1′] With $\beta^{(t)}$ and $\sigma_u^{2(t)}$, compute the working "response" vector, $\tilde{Z}$, and diagonal weight matrix, $\tilde{W}$, via

$$\tilde{Z} = X\beta^{(t)} + H^{-1}\left[y - h\left(X\beta^{(t)}\right)\right] \quad , \quad \tilde{W}_{ii} = \sigma_u^{2(t)} + \frac{a_i V\left(h\left(x_i^{\mathrm{T}}\beta^{(t)}\right)\right)}{\left[h'\left(x_i^{\mathrm{T}}\beta^{(t)}\right)\right]^2} \quad ,$$

where $H = diag\left[h'\left(x_i^{\mathrm{T}}\beta^{(t)}\right)\right]$.

2′] Compute new estimates via $\beta^{(t+1)} = (X^{\mathrm{T}}\tilde{W}^{-1}X)^{-1}X^{\mathrm{T}}\tilde{W}^{-1}\tilde{Z}$ and

$$\sigma_u^{2(t+1)} = \frac{\sum_i [y_i - h(x_i^{\mathrm{T}}\beta^{(t+1)})]^2 - \sum_i a_i V(\mu_i^{*(t+1)})}{\sum_i [h'(x_i^{\mathrm{T}}\beta^{(t+1)})]^2} \quad .$$

3′] Go to [1′] using $\beta^{(t+1)}$ and $\sigma_u^{2(t+1)}$ until convergence for both β and σ_u^2 .

A starting value of 0 for σ_u^2 leads to beginning the IWLS the same as for an ordinary GLM in Section 2.1.1. It has been suggested by some authors (McCullagh and Nelder, 1989 Ch. 14) that updating σ_u^2 need not be done at each iteration. Instead, let $\beta^{(t)}$ converge for a fixed estimate of σ_u^2, then update to a new estimate of σ_u^2 with above formulae and re-estimate β, going back and forth until both β and σ_u^2 estimates converge.

2.2.4 REML Estimation

Williams (1982) also mentioned an estimation method that has later been used to obtain restricted maximum likelihood-*type* (REML) estimates (Schall, 1991). Williams claimed that this seems appropriate when the conditional distribution of Y is adequately approximated by a normal curve (i.e., when the binomial index m is large or the Poisson counts are large). The idea is to iteratively apply a linearized form of the link to the response data and assume that this transformed data follows the normal linear mixed model assumptions.

To a first order approximation, $g(Y) \simeq g(\mu) + g'(\mu)^{\mathrm{T}}(Y - \mu)$. Define a new working dependent variate Z^* as being equal to the right hand side, i.e.,

$$Z^* = X\beta + u + g'(\mu)^{\mathrm{T}}[Y - h(X\beta + u)] \quad . \tag{2.5}$$

The idea is that if $g'(\mu)^{\mathrm{T}}[Y - h(X\beta + u)]$, the "experimental error", is assumed normal with constant variance, then (2.5) is a normal linear mixed model. Due to the robustness of least squares to normality departures, the methods of REML (Harville, 1977, Patterson and Thompson, 1971) for the linear mixed model should yield good estimates and inferences, asymptotically. Schall (1991) demonstrates the application of normal linear mixed model methods of Fellner (1986, 1987) and REML methods of Patterson and Thompson (1971) to this model, leading to the IWLS scheme detailed below. Gilmour, Anderson and Rae (1985) also present this approach for the binomial probit model. Breslow and Clayton (1993) further link this to Green's (1987) Penalized Quasi-likelihood, to the empirical Bayes method of Stiratelli, Laird and Ware (1984), and to an approximation to the Laplace approximation (Liu and Pierce, 1993).

A connection between IGLS and REML can be drawn. From (2.5) computing the marginal moments for Z^* gives

$$\mathrm{E}\{Z^*\} = X\beta \quad \text{and} \quad \mathrm{Var}\{Z_i^*\} \simeq \sigma_u^2 + \frac{a_i V\Big(h\big(x_i{}^{\mathrm{T}}\beta^{(t)}\big)\Big)}{\Big[h'\big(x_i{}^{\mathrm{T}}\beta^{(t)}\big)\Big]^2} \quad .$$

These are exactly the mean and variance relationship for the working variate $z_i^{(t)}$ from (2.4). Thus IGLS may be considered an approximation to the REML-*type* estimation method.

In the IWLS scheme below, the u's are considered as parameters to be estimated, from which an estimate for σ_u^2 can be obtained (Schall, 1991; Breslow and Clayton, 1993).

1″] With $\beta^{(t)}$, $u^{(t)}$, and $\sigma_u^{2(t)}$, compute the working "response" vector, $\tilde{Z}$, and diagonal matrix $\tilde{W}$, via

$$\tilde{Z} = X\beta^{(t)} + \mathrm{u}^{(t)} + H^{-1}\Big[y - h\Big(X\beta^{(t)} + u^{(t)}\Big)\Big] \quad , \quad H = diag\Big[h'\Big(x_i{}^{\mathrm{T}}\beta^{(t)} + u^{(t)}\Big)\Big] \quad ,$$

$$\tilde{W}_{ii} = \sigma_u^{2(t)} + \frac{a_i V\Big(h\Big(x_i{}^{\mathrm{T}}\beta^{(t)} + u^{(t)}\Big)\Big)}{\Big[h'\Big(x_i{}^{\mathrm{T}}\beta^{(t)} + u^{(t)}\Big)\Big]^2} \quad .$$

2″] Compute new estimates $\beta^{(t+1)}$, $u^{(t+1)}$, and $\sigma_u^{2(t+1)}$ via

$$\beta^{(t+1)} = (X^{\mathrm{T}}\tilde{W}^{-1}X)^{-1}X^{\mathrm{T}}\tilde{W}^{-1}\tilde{Z} \quad , \quad u^{(t+1)} = \sigma_u^{2(t)}\tilde{W}^{-1}(\tilde{Z} - X\beta^{(t+1)}) \quad ,$$

$$\sigma_u^{2(t+1)} = \frac{1}{n-\alpha}\sum_i (u_i^{(t+1)})^2 \text{ , where } \alpha = \frac{1}{\sigma_u^{2(t)}} tr(\Gamma_{22}) \text{ and } \Gamma_{22} \text{ is from}$$

$$\begin{bmatrix} \Gamma_{11} & \Gamma_{12} \\ \Gamma_{21} & \Gamma_{22} \end{bmatrix} = \begin{bmatrix} X^{\mathrm{T}}\tilde{W}^{-1}X & X^{\mathrm{T}}\tilde{W}^{-1} \\ \tilde{W}^{-1}X & \tilde{W}^{-1} + D^{-1} \end{bmatrix}^{-1} \quad \text{with } D = \mathrm{Cov}(u) \quad .$$

3″] Go to [1″] using $\beta^{(t+1)}$, $u^{(t+1)}$, and $\sigma_u^{2(t+1)}$ until convergence for β and σ_u^2.

2.2.5 Comments About Quasi-likelihood Estimation

Quasi-likelihood (QL) methodology can be used for the estimation of β in (2.3) by incorporating all random effects into a single term. This is a rather simplistic approach, but very convenient since most statistical software packages already provide QL estimation. Ignoring the random effects and treating the data as essentially an ordinary GLM typically will result in overdispersion; that is, the data will exhibit more variation than expected, relative to a specified probability distribution. Quasi-likelihood was developed as a means for modeling the overdispersion as a single scale parameter. Wedderburn's (1974) QL

methodology essentially lets ϕ be "unknown" in the specification of the GLM (see Section 2.1.1). The connection between QL and (2.4) is simply that QL replaces the marginal variance of y_i in (2.4) with $\phi a_i V(\mu_i^*)$.

The use of QL for more than one random effect is done with no changes. Thus, as will be seen in Section 2.3, the induced correlations among observations sharing the same random component would be totally ignored. If this correlation is large, it is known from linear models that resulting t- and F-tests can become severely optimistic or conservative, depending on the "sign" of the correlation. This result might be expected to carry over to GLM.

2.3 Estimation for Generalized Linear Models with Two Random Effects Added to the Fixed Effects

Recall the model presented in Section 1.2 for modeling two random effects added to the fixed effects. Let Y represent an n by 1 vector of response variables, X an n by k matrix of fixed explanatory variables, and take u_1 and u_2 to be r_1 by 1 and r_2 by 1 random vectors which are independent of each other. Suppose that given u_1 and u_2, y_i follows a one-parameter exponential family distribution:

$$f(y_i|(A_1u_1)_i,\ (A_2u_2)_i;\ \theta_i) = exp\{[y_i\theta_i - b(\theta_i)]/a_i + c(y_i, a_i)\}\ ,$$

where A_1 and A_2 are the known n by r_1 and n by r_2 incidence matrices for the random effects. Suppose further that $u_j \sim N_{r_j}(\mathbf{0},\ \sigma_j^2 \boldsymbol{I})$, $j = 1,2$ and that

$$\mathrm{E}\{Y|u_1,\ u_2\} = \mu\ \ ,\ \text{with}\ g(\mu) = X\beta + A_1u_1 + A_2u_2\ \ .$$

For convenience, let A=$[A_1\ A_2]$ and u=$[u_1^{\mathrm{T}}\ u_2^{\mathrm{T}}]^{\mathrm{T}}$. Generalizing to q random effects is done by adding them in the link in the obvious manner (i.e., letting A=$[A_1\ A_2\ \ldots\ A_q]$ and $u = [u_1^{\mathrm{T}}\ u_2^{\mathrm{T}}\ \ldots\ u_q^{\mathrm{T}}]^{\mathrm{T}}$). The link may be written as $g(\mu) = X\beta + Au$. For the model discussed in Section 2.2, q would be taken to be 1 and $A = I$, the identity matrix. The marginal density of Y can be written as:

$$f(Y;\beta,\underline{\sigma}^2) = |D|^{-1/2}\int exp\left\{\sum_{i=1}^{n}\frac{y_i\theta_i - b(\theta_i)}{a_i} - \tfrac{1}{2}u^{\mathrm{T}}D^{-1}u\right\}\mathrm{d}u\ \ , \qquad (2.6)$$

where $\underline{\sigma}^2 = [\sigma_1^2\ ,\ \ldots\ ,\ \sigma_q^2]^{\mathrm{T}}$, θ is a known function of β and u, D is the covariance

matrix of u, and the integration is over the general space spanned by the vector u (i.e., $\mathbb{R}^r$ where $r = r_1 + \ldots + r_q$). Typically, D is diagonal with diagonal submatrices $\left\{\sigma_1^2 I_{r_1}, \sigma_2^2 I_{r_2}, \cdots, \sigma_q^2 I_{r_q}\right\}$.

To demonstrate the matrix assignments, recall the cell irradiation data example in Chapter 1, where there were 9 independent trials and 3 dishes per trial. In this case, A_1 would be a 27 by 9 matrix with each column having 3 ones and 24 zeros, the ones indicating which trial the count was from, and A_2 would be a 27 by 27 identity matrix. Thus $r_1 = 9$, $r_2 = 27$ and D would be a 36 by 36 diagonal matrix with σ_1^2 in the first 9 elements and σ_2^2 in the last 27 elements of the main diagonal.

$$A_1^{\mathrm{T}} = \begin{bmatrix} 111000000000000000000000000 \\ 000111000000000000000000000 \\ 000000111000000000000000000 \\ 000000000111000000000000000 \\ 000000000000111000000000000 \\ 000000000000000111000000000 \\ 000000000000000000111000000 \\ 000000000000000000000111000 \\ 000000000000000000000000111 \end{bmatrix} \quad , \quad A_2 = I_{27} \quad , \quad D = \begin{bmatrix} \sigma_1^2 I_9 & 0 \\ 0 & \sigma_2^2 I_{27} \end{bmatrix} .$$

The methods for estimation from the previous section extend here, but with some complexities. When $A = I$, numerical integration reduces to the product of univariate integrals that can be evaluated simultaneously. Here, A is necessarily not I, and thus multi-dimensional numerical integration is required (and much more difficult than for a single dimension). The extra dispersion component forces marginal correlations among the components of Y, necessitating generalized rather than weighted least squares for methods based on moments. For these estimates, there are several choices for moment estimators of the dispersion components. Furthermore, there is no theory regarding inference about β that incorporates the uncertainty in estimation of the variance components—except to replace them by their estimates and treat the estimates of the regression parameters as approximately normal.

Details regarding estimation will be discussed in the following subsections, while inferences will be discussed in Section 2.4 . In Chapter 3, the Laplace approximation, as an extension of the work by Liu and Pierce (1993), will be presented. The goal there will be to provide likelihood based inferences regarding the fixed effects.

2.3.1 Maximum Likelihood

As can be seen, with more dispersion components there is substantial added complexity in the integral corresponding to (2.6). Some simplification can be made if random effects are nested, reducing the dimension of integration some, but multidimensional integration is still required. For numerical integration methods, the additional random effects increase the number of functional evaluations in an exponential fashion, quickly becoming too large for practical purposes.

For a general case, the Gauss-Hermite numerical integration of (2.6) would be given by:

$$\frac{1}{\sqrt{2}} \sum_{l_1=1}^{m_1} \sum_{l_2=1}^{m_2} \cdots \sum_{l_r=1}^{m_r} \left[\prod_{j=1}^{r} w_{l_j} \right] F\left(\sqrt{2} D^{1/2} v\right) ,$$

where w_{l_j} are standard weights, $v = [v_{l_1}\ v_{l_2} \ldots v_{l_r}]^{\mathrm{T}}$ with v_{l_j} being standard nodes, $D^{1/2}$ denotes Cholesky matrix decomposition, and F is the conditional density of Y given u, considered a function of u. Typically, $m_1 = m_2 = \cdots = m_r = m$, resulting in m^r functional evaluations of F. Estimation is carried out applying a numerical grid search.

The EM algorithm is virtually unchanged from that presented in Section 2.2.2. The integrals, however, are now required to be multidimensional. Andersen and Aitkin (1985) presents an iterative weighted least squares scheme for the single random effects case for binomial data, along with needed modifications for extension to 2 nested random effects. The use of the Laplace approximation to the integral in (2.6) and the subsequent maximum likelihood analysis are deferred here and discussed in Chapter 3.

2.3.2 Estimation Based on Moment Assumptions

In Section 2.2.3, an IWLS routine was developed for the single random effect when $A = I$. For the case $A \neq I$, (2.4) must be altered to have a non-diagonal covariance structure. For the general case, write the nonlinear model as

$$Y = h(X\beta + Au) + \epsilon \quad ,$$

where $h(\ ) = g^{-1}(\ )$, ϵ has mean 0 and variance $a_i V(\mu_i)$, $\mu_i = h(x_i^{\mathrm{T}}\beta + A_i^{\mathrm{T}}u)$, and A_i^{T} is the i^{th} row of A. Expanding $h(\ )$ about $u = 0$ leads to the approximate conditional moments model, assuming $|D|$ is small:

$$Y \simeq h(X\beta) + HAu + \epsilon^* \quad ,$$

where $H = diag\left[h'(x_i{}^{\mathrm{T}}\beta)\right]$, the ϵ^*'s are independent random errors with mean 0 and variance matrix V, $V_{ii} = a_i V(\mu_i^*)$, $\mu^* = h(X\beta)$. This yields the approximate marginal mean vector and covariance matrix

$$\mathrm{E}\{Y\} \simeq \mu^* = h(X\beta) \ \text{ and } \ \mathrm{Cov}\{Y\} \simeq V + HADA^{\mathrm{T}}H \ .$$

These are the same moments from (2.4) except that the covariance matrix is not diagonal— off–diagonal components reflect induced correlations. Applying a second linearization, this time in β, yields at the t^{th} iteration

$$Z^{(t)} = X\beta + \epsilon^*$$

where

$$Z^{(t)} = H^{-1}\left[y - h(X\beta^{(t)})\right] + X\beta^{(t)} \quad , \quad H = diag\left[h'\left(x_i{}^{\mathrm{T}}\beta^{(t)}\right)\right]$$

and

$$\mathrm{E}\{\epsilon^*\} = 0 \ , \quad \mathrm{Cov}\{\epsilon^*\} = ADA^{\mathrm{T}} + H^{-1}\mathrm{V}H^{-1}.$$

Notice that the mean is modeled exactly the same as an ordinary GLM ignoring the random effects altogether; and that these moments are identical to those resulting from (2.4) for the single random effect model, with $\sigma_u^2 \boldsymbol{I}$ generalized to ADA^{T} to reflect induced correlations among observations having the same random component. Because some of the off–diagonal elements of $\mathrm{Cov}\{\epsilon^*\}$ are non-zero, generalized least squares (rather than weighted least squares) is used. Hence, this method has been denoted throughout this thesis as the iterative generalized least squares (IGLS) method. Estimation of the dispersion components may be accomplished by method of moments (McCullagh and Nelder, 1989 Ch. 14; Morton, 1987; Firth and Harris, 1991), but several choices are available.

This IGLS scheme requires only simple modifications to the IWLS presented in Section 2.2.3. In step [1′] replace the equation for $\tilde{W}$ by

$$\tilde{W} = A\tilde{D}A^{\mathrm{T}} + H^{-1}\tilde{V}H^{-1} \quad ,$$

where $\tilde{D}$ and $\tilde{V}$ use current estimates of β and the dispersion components, and where $H = diag\left[h'\left(x_i{}^{\mathrm{T}}\beta^{(t)}\right)\right]$. The estimation of dispersion components is done in step [2′], using method of moments (MOM) equations. There are several possible MOM equations, depending on which set of quadratic forms is used. The choice used here is discussed below. After $\tilde{\mathrm{D}}$ is updated, go back to [1′] until estimates of β and the dispersion components converge.

It is unclear from the literature as to which set of quadratic forms should be used in obtaining MOM estimates. If there are many levels of each random effect (i.e., r_j are large for $j = 1,\ldots,q$) then it should not matter which set is used since all should give similar estimates for D. In fact, it is felt that any consistent estimate of D can be used without affecting the estimation of β, as suggested by the work of Liang and Zeger (1986), Zeger and Liang (1986) and Prentice (1988). For the numerical work in this dissertation, the intuitive set involving projection operators as discussed in McCullagh and Nelder (1989, Ch. 14) are used.

In all of the designs in mind for this dissertation, the approximate marginal covariance matrix can be partitioned into the form

$$\mathrm{Cov}\{Y\} = V_0 + \sigma_1^2 V_1 + \ldots + \sigma_q^2 V_q \quad ,$$

where V_0 is diagonal with elements V_{ii}, and $V_1,\ldots,V_q$ involve known functions of β, typically of the form $HA_j(A_j^{\mathrm{T}}A_j)^{-1}A_j^{\mathrm{T}}H$ for $j = 1,\ldots,q$ (where A_j is the n by r_j incidence matrix for random effect u_j) and $H = diag\left[h'(x_i{}^{\mathrm{T}}\beta)\right]$. A set of q quadratic forms was chosen by taking

$$Q_j = [y - h(X\beta^{(t)})]^{\mathrm{T}}A_j(A_j^{\mathrm{T}}A_j)^{-1}A_j^{\mathrm{T}}[y - h(X\beta^{(t)})], \quad j = 1,\ldots,q \ .^{1}$$

From these, q equations in the q unknown σ_j^2's were formed by noting that

[1]To allow for conditional overdispersion as in Schall (1991), V_0 would be replaced by ϕV_0 and $Q_0 = (y - h(X\beta^{(t)}))^{\mathrm{T}}(y - h(X\beta^{(t)}))$ would be introduced, leading to $q+1$ quadratic forms.

$$\mathrm{E}\{Q_j\} = \sum_{l=1}^{q} \mathrm{tr}(A_j(A_j^{\mathrm{T}}A_j)^{-1}A_j^{\mathrm{T}}V_l)\sigma_l^2 + \mathrm{tr}(A_j(A_j^{\mathrm{T}}A_j)^{-1}A_j^{\mathrm{T}}V_0) \ .$$

Dispersion component estimates can then be obtained, updating $\tilde{D}$ in step [2′].[2]

Of course, the estimates of the dispersion components are not guaranteed to be non-negative. Some checking must be done to ensure this, with corrections made to other components when one or more are negative. The solution to this boundary-crossing problem used for results presented herein has been to reset the negative components to zero and re-estimate the rest under such a constraint. The properties of such a method are not known. This IGLS method is a straightforward extension of the IWLS methods suggested by Williams (1982) and Breslow (1984) (Section 2.2.3). The results here generalize those of Morton (1987) and Firth and Harris (1991) even though their conditional model is different. The IGLS method was included in Chapter 14 of McCullagh and Nelder (1989) where estimates for the dispersion components making up D are found via method of moments. Goldstein (1991) and Breslow and Clayton (1993) present this approach, with the latter paper comparing the result to those using REML (see Section 2.2.4) and suggests using the pseudolikelihood method (Carroll and Rupert, 1982) for estimating the dispersion components. The generalized estimating equations (GEE) method for similarly defined models of longitudinal studies employs IGLS for a particularly chosen correlation matrix (see Zeger and Liang; 1986, Section 3).

2.3.3 REML Estimation

The extension to two random effects involves a simple extension to the ideas in Section 2.2.4. Applying the linearized link to the data yields the approximate linear mixed model

$$Z^* = X\beta + Au + g'(\mu)^{\mathrm{T}}[Y - h(X\beta + Au)] \ ,$$

with estimation of β, u, and D done similarly to the IWLS scheme in Section 2.2.4. In Step [1″], replace $\tilde{W}$ by

[2]In cases where there is balance in the random effects, with balance defined by the column sums of each A_j being equal to a_j, $A_jA_j^{\mathrm{T}} = a_jA_j(A_j^{\mathrm{T}}A_j)^{-1}A_j^{\mathrm{T}}$, fewer matrix inversions are required.

$$\tilde{W} = A\tilde{D}A^{\mathrm{T}} + H^{-1}\tilde{V}H^{-1} \quad ,$$

where $H = diag\Big[h'\Big(x_i{}^{\mathrm{T}}\beta^{(t)} + A_i^{\mathrm{T}}u^{(t)}\Big)\Big]$, $\tilde{V} = diag\Big[a_i V\Big(h\Big(x_i^{\mathrm{T}}\beta^{(t)} + A_i^{\mathrm{T}}u^{(t)}\Big)\Big)\Big]$, and A_i^{T} is the i^{th} row of A. In Step [2″], the changes are given by:

$$u^{(t+1)} = \tilde{D}A^{\mathrm{T}}\tilde{W}^{-1}(\tilde{Z} - X\beta^{(t+1)})$$

and

$$\sigma_j^{2(t+1)} = \frac{u_j^{\mathrm{T}}u_j}{r_j - \alpha_j} \quad ,$$

where $\alpha_j = \mathrm{tr}(\Gamma_{jj})/\sigma_j^{2(t)}$, with Γ_{jj} being the r_j by r_j sub-matrix of T_{22} created from rows and columns numbered $r_1 + \ldots + r_{j\text{-}1} + 1$ through $r_1 + \ldots + r_{j\text{-}1} + r_j$ (i.e., those same rows and column numbers in D with dispersion component σ_j^2 on the diagonal) and where T_{22} is from the partitioned matrix (see Schall, 1991):

$$\begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} = \begin{bmatrix} X^{\mathrm{T}}\tilde{W}^{-1}X & X^{\mathrm{T}}\tilde{W}^{-1}A \\ A^{\mathrm{T}}\tilde{W}^{-1}X & A^{\mathrm{T}}\tilde{W}^{-1}A + \tilde{D}^{-1} \end{bmatrix}^{-1} \quad .$$

The α_j in step [2″] can be thought of as effective loss of degrees of freedom for estimating β. That these estimates yield REML shrinkage estimates is discussed in Schall (1991). Breslow and Clayton (1993) propose dispersion component estimation to be carried out using adjusted profile likelihoods (Cox and Reid, 1987). REML results presented in this thesis follow Schall's approach.

Schall (1991) presented this REML approach for the cell irradiation and salamander mating example data sets discussed in Chapter 1. He also allowed the conditional variance to be proportional to the assumed binomial variance (in the same way that QL models the marginal variance). Schall used as the conditional model

$$y_i = h(x_i^{\mathrm{T}}\beta + A_i^{\mathrm{T}}u) + \epsilon_i \quad , \quad \mathrm{Var}(\epsilon_i) = \phi a_i V(\mu_i).$$

Thus, in addition to the dispersion components σ_1^2 and σ_2^2, ϕ must also be estimated. Schall uses the linear mixed model REML estimating equations of Patterson and Thompson (1971) as if ϕ were the "experimental" error term (see Step 2′ of p. 722, Schall, 1991). For the cell irradiation data, Schall estimated ϕ to be 0.937 (basically 1) implying model (2.6) seems appropriate without an

extra term ϕ. Breslow and Clayton (1993) develop their approximate Laplace using a similarly defined log-quasilikelihood rather than the strict likelihood assumed in (2.6). Allowing a parameter ϕ in the conditional likelihood leading to (2.6) can be incorporated into (2.6) with little difficulty, but has not been done since the point here is to understand precision losses when (2.6) is correct.

2.4 Inferences about β in the Presence of Random Effects

Although there has been a great deal of interest in the past few years regarding GLM with Random Effects, the focus has been on estimation techniques. Three prominent methods in the literature have been IGLS, REML, and ML via numerical integration or EM algorithm (with ML being shied away from due to the lengthy computing times required). In this dissertation, QL has been included among these three as a naive approach, due to its ease in implementation and its availability in many statistical software packages. Very little has been done regarding inferences of the fixed effects parameters. Although it is possible to make inferences about the dispersion components, it is felt that rather large studies must be carried out in order to draw any meaningful inferences regarding them (this is also the case for the linear mixed model setting). Each of the three moments methods (QL, IGLS, and REML) have some asymptotic justification regarding inferences, yet little has been done to verify the results for small samples.

McCullagh (1983) and McCullagh and Nelder (1989, Section 9.2) present asymptotic theory for drawing inferences from the QL model. For inferences, it is assumed that

$$\hat{\beta} \sim N(\beta, \hat{\phi}(X^{\mathrm{T}}\tilde{W}^{-1}X)^{-1}) \ ,$$

where

$$\tilde{W}_{ii} = \frac{a_i V\big(h(x_i^{\mathrm{T}}\hat{\beta})\big)}{\big[h'(x_i^{\mathrm{T}}\hat{\beta}\big]^2} \ , \quad \hat{\phi} = \frac{1}{n-k}\sum_{i=1}^{n}\frac{[y_i - h(x_i^{\mathrm{T}}\hat{\beta})]^2}{a_i V(h(x_i^{\mathrm{T}}\hat{\beta}))} \ .$$

From this, approximate hypothesis tests and confidence intervals may be obtained from normal theory. If model (2.6) is true, then it was shown that there is induced correlations among observations with common u's. Thus the

asymptotic results for QL, which are based on assuming independent observations, are likely to give poor inferences.

McCullagh and Nelder (1989, section 9.3) give some asymptotic theory for inferences when there is dependence among the responses in the form $\text{Cov}\{Y\} = \sigma_u^2 V(\mu)$ where $V(\mu)$ is a matrix of known functions of the β's. For inferences from the IGLS model, it is assumed that

$$\hat{\beta} \sim N(\beta, (X^{\mathrm{T}}\tilde{W}^{-1}X)^{-1}) \quad ,$$

where

$$\tilde{W} = A\tilde{D}A^{\mathrm{T}} + H^{-1}\tilde{V}H^{-1} \ , \quad H = diag\Big[h'\Big(x_i{}^{\mathrm{T}}\hat{\beta}\Big)\Big] \ ,$$

$$\tilde{V}_{ii} = a_i V(h(x_i^{\mathrm{T}}\hat{\beta})) \quad \text{and} \quad \tilde{D} = \{\hat{\sigma}_1^2 I,\ldots,\hat{\sigma}_q^2 I\} \quad .$$

Tests and confidence intervals are made using normal theory. It is felt by many authors that estimation of β is robust to misspecification of the variance. McCullagh and Nelder (1989, p. 438) state that the estimate of β is fairly insensitive to how D (=Cov$\{u\}$) is estimated. But there is a concern regarding inferences, as McCullagh and Nelder go on to state that the resulting covariance matrix for $\hat{\beta}$ does depend on how well one models the variance and, therefore, some care is needed regarding the choice of quadratic forms used to update D at each iteration. No work has been done to investigate this, although this need has been recognized by others (Morton, 1987; Firth and Harris, 1991).

Inferences based on REML estimates also follow from the approximate normality of the estimate of β. The main assumption is that on the linearized scale, there is additivity among all effects. This simply requires that the correct link function be chosen. From there, standard results for inferences from linear mixed models justifies the normality assumptions for testing. Due to the known robustness of linear model inferences (e.g., Gauss-Markov Theorem), it is assumed to carry over here as well. Thus it is assumed that

$$\hat{\beta} \sim N(\beta, (X^{\mathrm{T}}\tilde{W}^{-1}X)^{-1}) \quad ,$$

where

$$\tilde{W} = A\tilde{D}A^{\mathrm{T}} + H^{-1}\tilde{V}H^{-1} \ , \quad H = diag\Big[h'\Big(x_i{}^{\mathrm{T}}\hat{\beta} + \hat{u}\Big)\Big] \ ,$$

$$\tilde{V}_{ii} = a_i V(h(x_i^{\mathrm{T}}\hat{\beta} + A_i\hat{u})) \quad \text{and} \quad \tilde{D} = \{\hat{\sigma}_1^2 I,\ldots,\hat{\sigma}_q^2 I\} \quad .$$

All three of the inference procedures above rely on asymptotic normality of $\hat{\beta}$. This itself imposes a quadratic, symmetric distribution about $\hat{\beta}$, which may or may not be the case for small data sets. Conversely, inferences may be obtained from profile likelihoods with no imposed shape placed on the distribution of $\hat{\beta}$. (One can view asymptotic normality as imposing a quadratic, symmetric shape on the profile likelihood). To obtain a profile likelihood for a particular parameter, that parameter is considered constant while the others are optimized in the reduce parameter space, with the optimum likelihood's value retained. This is repeated for several constant values of the parameter of interest. After a grid of points has been used, the resulting functional values of the likelihood are plotted versus the parameter's values, yielding a profile likelihood plot. For inferences, a log relative profile likelihood plot (Aitkin, Anderson, Francis and Hinde, 1991, section 2.15) is drawn, along with percentiles from the χ^2_1 distribution (the asymptotic distribution of the likelihood ratio statistic). Inferences for ML using Gauss-Hermite quadrature and the proposed solution based on the Laplace approximation will be carried out using log relative profile likelihoods. Details for obtaining confidence intervals from the log relative profile likelihood plot will be given in Chapter 3.

3. APPROXIMATE LIKELIHOOD ANALYSES USING THE LAPLACE METHOD

In Chapter 2, it was shown that numerical integration of the marginal density (2.5) was required to obtain MLEs and likelihood based inferences. The major drawback to this is the computational effort required to approximate, and optimize, the likelihood function. Although methods based on weaker assumptions are useful, the premise here is that likelihood methods are sometimes appropriate, that likelihood–based tests and confidence intervals are considerably better than those based on the moment methods in these cases, and that the computational burden can be substantially reduced by use of the Laplace approximation (presented in the next section). The resulting approximate likelihood still must be optimized numerically, but this can be accomplished easily with modern optimization routines.

The Laplace approximate likelihood function is presented in Section 3.1. In Section 3.2, its use for obtaining approximate MLEs and likelihood inferences will be detailed. Section 3.3 will discuss the adequacy of the approximation, demonstrating that, for practical purposes, exact MLEs and inferences are obtained.

3.1 The Laplace Approximation

The likelihood function of (2.6) may be re-written as (up to a known proportional constant)

$$L(\beta, \underline{\sigma}^2) = f(Y;\beta, \underline{\sigma}^2) = |D|^{-1/2} \int e^{G(u)}\, du \quad , \tag{3.1}$$

where

$$G(u) = \sum_{i=1}^{n} \frac{y_i\theta_i - b(\theta_i)}{a_i} - \tfrac{1}{2}u^{\mathrm{T}}D^{-1}u \quad \text{and } \underline{\sigma}^2 = [\sigma_1^2 \ldots \sigma_q^2]^{\mathrm{T}} \quad .$$

Applying the Laplace method for integral approximation (DeBruijn, 1961 Ch 4) yields the approximate likelihood function

$$L(\beta, \underline{\sigma}^2) \simeq |D|^{-1/2} \left| -G''(\tilde{u}) \right|^{-1/2} e^{G(\tilde{u})} \tag{3.2}$$

where $\tilde{u}$ maximizes $G(\ )$ (i.e., $G'(\tilde{u}) = 0$) and $G''(\tilde{u})$ is the Hessian evaluated at $\tilde{u}$. Denote the right hand side of (3.2) by $L_{La}(\beta, \underline{\sigma}^2)$.

Notice that the Laplace approximation to $\int e^{G(u)}\, du$ amounts to replacing $G(u)$ by a second–order Taylor's approximation of $G(u)$ about its maximizing value, $\tilde{u}$. See also Davison, 1986; Tierney and Kadane, 1986; Thisted, 1988 p. 316; Tierney, Kass and Kadane, 1989; Barndorff-Nielsen and Cox, 1989 Ch. 3; Cox, Hinkley, Reid and Snell, 1991 Section 12.5; and Kass, Tierney and Kadane, 1990 for more discussion of this approximation in statistical problems.

Later in this chapter, it will be shown that the error in L_{La} (eq. 3.2) is roughly constant, leading to exact likelihood results. In Section 3.3, a more detailed development of the Laplace method will be presented, along with an investigation of the error.

3.2 Maximum Likelihood Estimates and Likelihood Ratio Inferences

Estimates of $(\beta, \underline{\sigma}^2)$ are obtained by maximizing L_{La} over the possible parameter values (i.e., the parameter space). Due to the intractability of L_{La}, numerical optimization is needed. For inferences, L_{La} is maximized in the reduced parameter space, with the parameter(s) of interest held constant, via numerical grid search as well. These ideas are detailed below in sections 3.2.2 and 3.2.3. The adequacy of the estimates and inferences depend on how well L_{La} approximates the true likelihood function. This in turn depends on how well the maximizing value, $\tilde{u}$, is estimated.

3.2.1 Finding the Maximizing Value, $\tilde{u}$

The adequacy of the Laplace approximation L_{La} relies on finding $\tilde{u}$, which maximizes $G(u)$ (Liu and Pierce, 1993). This is the "difficult" step in evaluating L_{La}, due to the nonlinearity in u of $G'(u)$. Nonlinear optimization methods, such as the Gauss-Newton or Newton-Raphson methods can be used to find $\tilde{u}$ fairly easily. The Newton-Raphson method for root finding, a fairly simple and straightforward procedure, is detailed below as it is applied to finding $\tilde{u}$ for particular values of β and $\underline{\sigma}^2$.

The objective is to find $\tilde{u}$ such that $\partial G(u)/\partial u$ evaluated at $\tilde{u}$ is 0, where

$$\partial G(u)/\partial u = \sum_{i=1}^{n} \frac{1}{a_i}\left(y_i \frac{\partial \theta_i}{\partial \eta_i}\frac{\partial \eta_i}{\partial u} - \frac{\partial b(\theta_i)}{\partial \theta_i}\frac{\partial \theta_i}{\partial \eta_i}\frac{\partial \eta_i}{\partial u}\right) - D^{-1}u$$

$$= \sum_{i=1}^{n} \frac{(y_i - \mu_i)h'(\eta_i)}{a_i V(\mu_i)} A_i^{\mathrm{T}} - D^{-1}u \quad ,$$

and $\eta_i = x_i^{\mathrm{T}}\beta + A_i^{\mathrm{T}}u$, $\mu_i = h(\eta_i)$, $V(\mu_i) = b''(\theta_i)$. This is conveniently written as

$$G'(u) = A^{\mathrm{T}}(Y - \mu)V^{-1}H^{-1} - D^{-1}u \quad ,$$

where V is diagonal with elements $a_i V(\mu_i)/[h'(\eta_i)]^2$ and $H = diag\big[h'(\eta_i)\big]$. In the case that $g(\ )$ is the canonical link, $G'(u)$ is simplified by noting that $V^{-1}H^{-1} = I$. Differentiate again with respect to u to get the Hessian matrix

$$G''(u) = -\sum_{i=1}^{n} \frac{[h'(\eta_i)]^2}{a_i V(\mu_i)} A_i^{\mathrm{T}} A_i - D^{-1} + R \quad ,$$

or, equivalently

$$= -A^{\mathrm{T}}V^{-1}A - D^{-1} + R \quad ,$$

with $R = \sum_{i=1}^{n} \frac{(y_i - \mu_i)}{a_i}\left[\frac{h''(\eta_i)}{V(\mu_i)} - \frac{h'(\eta_i)^2 b'''(\theta_i)}{V(\mu_i)^3}\right] A_i^{\mathrm{T}} A_i$.[3]

The Newton-Raphson root finding method yields the following iterative scheme for estimating $\tilde{u}$:

1] with $u^{(t)}$, the current guess of $\tilde{u}$, evaluate $G'(u^{(t)})$ and $G''(u^{(t)})$.

2] update $u^{(t+1)} = u^{(t)} - G''(u^{(t)})^{-1}G'(u^{(t)})$.

3] iterate until $\| u^{(t)} - u^{(t+1)} \| < tol$, where tol is some pre-specified convergence tolerance criteria and $\| \ \|$ denotes some norm.

The actual norm used during applications to the data examples and simulations

[3] When the canonical link is used, $h'(\eta_i) = b''(\theta_i) = V(\mu_i)$ and $h''(\eta_i) = b'''(\theta_i)$ yielding $R = 0$.

was a relative convergence norm

$$\left\| u^{(t)} - u^{(t+1)} \right\| = \max_{1 \le j \le r} \left| \frac{u_j^{(t)} - u_j^{(t+1)}}{u_j^{(t)}} \right|$$

with $tol = \frac{1}{10}\%$ and an initial guess of $u^{(0)} = 0$. Convergence problems occurred, seemingly at random, on various occasions, resulting from "poor" initial guesses. However, these were circumvented by placing a maximum step size of .5 in the Newton-Raphson procedure, which eliminated all divergence situations.[4] The number of iterations until convergence was reduced some by setting $u^{(0)}$ to the root of a linearized form of $G'(u)$—linearized in u. When the canonical link is used, this amounted to using the root of

$$A^{\mathrm{T}}\left[y - h(X\beta) - HAu\right] - D^{-1}u \quad , \text{ where } H = diag\left[h'(x_i^{\mathrm{T}}\beta)\right] \quad .$$

An obvious criticism with using Laplace is the need to compute $G''(u)^{-1}$ for estimation of $\tilde{u}$. However, when the random effects are nested, as in Example 1, this seems to be off little importance since $G''(u)$ will be block diagonal, and thus the inverse should not run into computer precision problems. But when the random effects are crossed, as in the Salamander Mating example, the Hessian of G can be very unstructured and therefore computer accuracy of its inverse is more questionable. Yet this problem is no different than what is encountered with IGLS and REML estimation where matrices denoted by $\tilde{W}$ must be inverted. And in REML this $\tilde{W}^{-1}$ is needed for computing submatrices of a much larger matrix to be inverted. Thus, if computer accuracy is an issue in using Laplace for obtaining $G''(u)^{-1}$, it is as well for IGLS and REML.

3.2.2 Optimization of the Approximate Likelihood Function

The approximate log-likelihood function for the assumed true model, based on the Laplace approximation, is

$$\ell_{La}(\beta, \underline{\sigma}^2) = -\tfrac{1}{2} ln(|D|) - \tfrac{1}{2} ln\left(\left| -G''(\tilde{u}) \right|\right) + G(\tilde{u}) \quad . \tag{3.3}$$

[4] Further refinement of the maximum step size needs to be addressed to allow more efficient use of the Newton-Raphson method here.

Differentiation of (3.3) with respect to β and $\underline{\sigma}^2$ is very difficult due to the determinants and the intractability of $\tilde{u}$ as a function of β and $\underline{\sigma}^2$. Therefore numerical methods are used to find the values that maximize it.

There are several possible grid search routines available with which to numerically find a maximum of ℓ_{La}. These include Nelder-Mead (Nelder and Mead, 1965; Kotz, Johnson and Read, 1985 Vol. 6 pp. 178-181), Quasi-Newton (Shanno, 1970), Sequential Quadratic Programming (Powell, 1978), and a full grid evaluation. The full grid search is practical only with a single β and single σ^2, while the others are intended to be "smarter" grid searches using information about the shape of ℓ_{La} obtained from previous evaluations. The particular routine used for the example data sets and simulations included in this dissertation has been the Nelder-Mead Simplex Algorithm. This routine is available in the programming language MatLab (The MathWorks, Inc., 1990) and is known to be a robust procedure with regards to the shape of the function of interest. The Quasi-Newton method requires numerical differentiation, a time consuming method with potentially poor results for non-smooth functions. Sequential Quadratic Programming is designed to handle constrained optimization and may be a good alternative procedure.

The Nelder-Mead simplex algorithm obtains points of optimization by surrounding the current guess by a simplex (a triangle in two-dimensions, a cube in three-dimensions, and so forth). Based on the functional values at the vertices, the simplex is redrawn with one vertex replaced by a vertex in the direction towards a better functional value. The simplex moves towards the optimum, then shrinks once the optimum is "surrounded". The centroid of the final simplex is the guess of the optimum. This approach is not very fast, yet is robust to non-smooth functions.

To avoid the potential problem presented by using the Nelder-Mead Simplex Algorithm in a constrained optimization problem (the dispersion components must be non-negative), the dispersion components were optimized on the natural logarithm scale. The Nelder-Mead algorithm found in MatLab (The MathWorks, Inc., 1990) was modified to stop if a dispersion component's estimate was getting close to 0 (i.e., the estimate on the *ln* scale was large negative) to avoid computer precision problems. If all vertices of the current simplex had a common dispersion component very small ($ln\,\sigma_j^2 < -12$) then that

dispersion component, σ_j^2, was estimated and fixed at 0, the parameter space was reduced by one dimension, and the other parameters were estimated holding σ_j^2 at 0. This modification is reasonable provided the objective function is smooth.

In normal linear mixed models, the likelihood function is often smooth and "steep" in the fixed effects, but is commonly multimodal and flat for the dispersion components. It seems very reasonable to assume similar properties here. From the examples and simulations, this does appear to be the case. The simplex quickly converges for the fixed effects; but convergence in the dispersion components is very slow, with values of the objective function changing little (relative to changes when a β value is changed) from guess to guess. The Nelder-Mead algorithm was at times re-started at the MLEs and took nearly as many iterations to converge as the original MLE optimization, yet would not necessarily yield the same MLEs —the fixed effects parameters would be unchanged, but the dispersion component estimates might be quite different yet yielding the same value of the objective function. This is not unlike the linear mixed model situation, where the likelihood may be multi-modal for the variance components. For these reasons, it is felt that estimates and inferences for the dispersion components are not reliable, yet reasonable estimates and inferences for the fixed effects are given.

One drawback to using the Nelder-Mead Simplex Algorithm for the numerical optimization is that its use has typically been limited to 10 or fewer parameters. And one can easily imagine a case with more than 10 parameters, especially if treatments are in a factorial structure. However, many studies will not exceed 10 parameters. In cases were they do, other numerical optimization routines which have demonstrated usefulness in large parameter settings should be considered.

3.2.3 Inferences Based on the Profile Likelihood

The generalized likelihood ratio test statistic for the hypothesis $\beta = \beta_0$ is given by

$$\lambda = -2[\ell(\beta_0, \tilde{\underline{\sigma}}^2_{\beta_0}) - \ell(\hat{\beta}, \hat{\underline{\sigma}}^2)] \quad ,$$

where $(\hat{\beta}, \underline{\hat{\sigma}}^2)$ denotes the MLEs and $\underline{\tilde{\sigma}}^2_{\beta_0}$ denotes the vector of dispersion components which maximize the likelihood when β is set to the hypothesized value, β_0. The hypothesis is rejected at level α if

$$\lambda > \chi^2_{\alpha,k} \quad ,$$

where $\chi^2_{\alpha,k}$ is the $1-\alpha$ percentile of the χ^2 distribution with k, the number of parameters in β, degrees of freedom. If the fixed effects are partitioned as $\{\beta, \gamma\}$ and the hypothesis is of the form $\beta = \beta_0$, γ unconstrained, then the test is formed by

$$\lambda = -2[\ell(\beta_0, \tilde{\gamma}_{\beta_0}, \underline{\tilde{\sigma}}^2_{\beta_0}) - \ell(\hat{\beta}, \hat{\gamma}, \underline{\hat{\sigma}}^2)] \quad ,$$

and the hypothesis is rejected if $\lambda > \chi^2_{\alpha,k}$, where k is the number of parameters in β. Equivalently, the hypothesis is rejected if

$$\ell(\beta_0, \tilde{\gamma}_{\beta_0}, \underline{\tilde{\sigma}}^2_{\beta_0}) < \ell(\hat{\beta}, \hat{\gamma}, \underline{\hat{\sigma}}^2) - \tfrac{1}{2}\chi^2_{\alpha,k} \quad . \tag{3.4}$$

Often, k is 1 and β is called the parameter of interest while γ and $\underline{\sigma}^2$ are labeled nuisance parameters.

A $(1-\alpha)100\%$ confidence set contains all values of β_0 such that a test of $\beta = \beta_0$ would fail to be rejected by an *α–level* test. Due to the continuity and convexity of the log-likelihood function, this confidence set is an ellipsoid (or an interval if $k = 1$). A *likelihood–based confidence set* (interval) for β would be all values of β_0 such that (3.4) does not hold. That is, those values of β_0 with log-likelihood values, $\ell(\beta_0, \tilde{\gamma}_{\beta_0}, \underline{\tilde{\sigma}}^2_{\beta_0})$, being no more than $\frac{1}{2}\chi^2_{\alpha,k}$ units from the maximum log-likelihood at the MLE.

The boundaries of the confidence set usually cannot be obtained analytically, but can be found by computing $\ell(\beta_0, \tilde{\gamma}_{\beta_0}, \underline{\tilde{\sigma}}^2_{\beta_0})$ over a grid of β_0 values, and finding boundaries by inspection for a specified α. When $k = 1$, Aitkin, Anderson, Francis and Hinde (1989, Sections 2.15, 4.5) suggest plotting

$$\ell(\beta_0, \tilde{\gamma}_{\beta_0}, \underline{\tilde{\sigma}}^2_{\beta_0}) - \ell(\hat{\beta}, \hat{\gamma}, \underline{\hat{\sigma}}^2) \quad \text{vs} \quad \beta_0$$

to create a *log relative profile likelihood* graph. From such a graph, any level α confidence interval can be found by taking as endpoints the two intersection points of the log relative profile likelihood curve with a horizontal line at $\frac{-1}{2}\chi^2_{\alpha,1}$.

The discouraging aspect in applying this methodology to (3.3) is that computing $(\tilde{\gamma}_{\beta_0}, \tilde{\underline{\sigma}}^2_{\beta_0})$, the maximizing parameter values in the reduced parameter space with β fixed at β_0, must be done by applying a grid search (typically the same used in getting MLEs) in the reduced parameter space. The convergence of the Nelder–Mead algorithm is slow, even in this reduced space. Recall that convergence was slow for the dispersion components. Since all dispersion components are still to be maximized, finding $(\tilde{\gamma}_{\beta_0}, \tilde{\underline{\sigma}}^2_{\beta_0})$ requires many iterations, even if it is started at the global MLEs. The number of iterations can be reduced somewhat by using well chosen starting values.

Aitkin, Anderson, Francis and Hinde (1989, p. 115) discuss the merits for preferring likelihood based inferences over those from assuming normality of the parameter estimates. The main reason is that the normality assumption forces a quadratic shape to the profile likelihood, which can be a very poor approximation.

3.3 Adequacy of the Laplace Approximate MLEs and Inferences

The exact log–likelihood function for β and $\underline{\sigma}^2$ can be written as

$$\ell(\beta, \underline{\sigma}^2) = -\tfrac{1}{2} ln|D| + ln\left(\int e^{G(u)} \mathrm{d}u\right) , \tag{3.5}$$

where $G(u)$ is defined as in (3.1). This integral in (3.5) can be re-written as

$$\begin{aligned}
\int e^{G(u)} \mathrm{d}u &= \int e^{G(u) + (1-1)G(\tilde{u}) + \frac{1}{2}(1-1)\{(u-\tilde{u})^{\mathrm{T}} G''(\tilde{u})(u-\tilde{u})\}} \mathrm{d}u \\
&= e^{G(\tilde{u})} \int s(u) e^{-\frac{1}{2}(u-\tilde{u})^{\mathrm{T}}[-G''(\tilde{u})](u-\tilde{u})} \mathrm{d}u \\
&= e^{G(\tilde{u})} \left| -G''(\tilde{u}) \right|^{-1/2} \left| -G''(\tilde{u}) \right|^{1/2} \int s(u) e^{-\frac{1}{2}(u-\tilde{u})^{\mathrm{T}}[-G''(\tilde{u})](u-\tilde{u})} \mathrm{d}u \\
&= e^{G(\tilde{u})} \left| -G''(\tilde{u}) \right|^{-1/2} \mathfrak{E}\{s(u)\} \quad ,
\end{aligned}$$

where $\tilde{u}$ is again the maximizing value of $G(u)$, $\mathcal{E}$ is an expectation function taken with respect to the distribution $MVN(\tilde{u},\ -G''(\tilde{u})^{-1})$, and

$$s(u) = exp\Big\{G(u) - G(\tilde{u}) - \tfrac{1}{2}(u-\tilde{u})^{\mathrm{T}}G''(\tilde{u})(u-\tilde{u})\Big\}\ .$$

Thus, the exact log–likelihood function is $\ell_{La}(\beta,\ \underline{\sigma}^2) + ln(\mathcal{E}\{s(u)\}$; and $\mathcal{E}\{s(u)\}$ can be thought of as the relative error in the Laplace approximation to the exact likelihood function.

If $\mathcal{E}\{s(u)\}$ is constant in β and $\underline{\sigma}^2$ (or nearly so), then the approximate MLEs and inferences based on the Laplace approximation will be identical (or nearly so) to the exact results. An investigation into the adequacy of the Laplace MLEs and inferences can be accomplished by studying $\mathcal{E}\{s(u)\}$.

3.3.1 Theoretical Results

First, consider expanding $s(u)$ about $\tilde{u}$ in a Taylor's expansion. This gives

$$s(u) \simeq s(\tilde{u}) + s'(\tilde{u})^{\mathrm{T}}(u-\tilde{u}) + \tfrac{1}{2}(u-\tilde{u})^{\mathrm{T}}s''(\tilde{u})(u-\tilde{u}) + \cdots$$

where

$$s(\tilde{u}) = 1\ \ ,\ \ s'(\tilde{u}) = 0\ \ ,\ \ s''(\tilde{u}) = 0\ \ ,\ \text{and } s^{(d)}(\tilde{u}) = G^{(d)}(\tilde{u}) \text{ for } d \geq 3\ .$$

Therefore $\mathcal{E}\{s(u)\} \approx 1$ and the error depends on third and higher moments of $(u-\tilde{u})$. Notice that this implies, with a 2^{nd} order approximation, the Laplace approximation (3.2) is exact. However the error from Taylor's series approximation says little about whether or not $\mathcal{E}\{s(u)\}$ is nearly constant over the parameter space, and hence sheds little light on the adequacy of the MLEs and inferences based on (3.2).

3.3.2 Numerical Results

Alternatively, $\mathcal{E}\{s(u)\}$ can be studied numerically with Gauss-Hermite quadrature. Consider, first, the case where u is a scaler. Then using standard Gauss-Hermite weights (w_l) and nodes (v_l) [Abromowitz and Stegun, 1972],

$$\mathcal{E}\{s(u)\} = (-G''(\tilde{u})/2\pi)^{\frac{1}{2}} \int e^{-\frac{1}{2}(u-\tilde{u})^2[-G''(\tilde{u})]} s(u)\mathrm{d}u$$

$$\simeq (-G''(\tilde{u})/2\pi)^{\frac{1}{2}} (-G''(\tilde{u})/2)^{-\frac{1}{2}} \sum_{l=1}^{m} w_l \, s\left(\sqrt{\frac{-2}{G''(\tilde{u})}}\, v_l + \tilde{u}\right)$$

$$= \frac{1}{\sqrt{\pi}} \sum_{l=1}^{m} w_l \, s\left(\sqrt{\frac{-2}{G''(\tilde{u})}}\, v_l + \tilde{u}\right).$$

If the number of nodes, m, is taken to be 1, then $w = \sqrt{\pi}$, $v = 0$, and $\mathcal{E}\{s(u)\} = s(\tilde{u}) = 1$. The general error term in Gauss-Hermite quadrature using m nodes is $\frac{m!}{2^m(2m)!} s^{(2m)}(\xi)$, $\xi \in \mathbb{R}$ (see Abromowitz and Stegun, 1972). This suggests that the error in the Laplace approximation is of the order of $\frac{1}{4}s''(\xi)$ for some $\xi \in \mathbb{R}$.

For u taken to be a vector, the Gauss-Hermite approximation is

$$\mathcal{E}\{s(u)\} \simeq \pi^{-r/2} \sum_{l_1=1}^{m_1} \sum_{l_2=1}^{m_2} \cdots \sum_{l_r=1}^{m_r} \left(\prod_{j=1}^{r} w_{l_j} \right) s\left((-G''(\tilde{u})/2)^{-1/2} \, v + \tilde{u} \right),$$

where $v = [v_{l_1} \; v_{l_2} \ldots v_{l_r}]^{\mathrm{T}}$ and $(-G''(\tilde{u})/2)^{-1/2}$ denotes Cholesky matrix decomposition. Typically, $m_1 = m_2 = \cdots = m_r = m$, resulting in m^r functional evaluations of s. For $m = 1$, $w_{l_j} = \sqrt{\pi}$ and $v_{l_j} = 0$, resulting in $\mathcal{E}\{s(u)\} \simeq s(\tilde{u}) = 1$ (i.e., the Laplace approximation is equal to 1-node Gaussian quadrature).

Some numerical evaluation of the adequacy of the Laplace approximation has been carried out with Gauss-Hermite quadrature on $\mathcal{E}\{s(u)\}$ using an example presented in Chapter 1 and some related simulated data sets.

3.3.2.1 Numerical Integration of the Relative Error for Cell Irradiation Data

Recall the Cell Irradiation example data set in Section 1.4.1. Conditional on the trial and dish effects being known, the observed counts are taken to be distributed as binomial with mean $400\pi_{ij}$ ($i = 1, \ldots, 9$; $j = 1, 2, 3$), with

$$ln[\pi_{ij}/(1-\pi_{ij})] = \beta + u_{1i} + u_{2ij} \quad , \text{ and } \quad u_{1i} \sim N(0, \sigma_1^2) \; , \; u_{2ij} \sim N(0, \sigma_2^2) \quad .$$

The log profile likelihood for β computed using the Laplace approximate likelihood is identical to that computed using Gauss-Hermite quadrature (see Figure 1).

In this case, $s(u)$ is defined to be

$$exp\Big\{y^{\mathrm{T}}A(u-\tilde{u}) - 1^{\mathrm{T}}[b(\eta_u) - b(\eta_{\tilde{u}})] + \tfrac{1}{2}(u-\tilde{u})^{\mathrm{T}}A^{\mathrm{T}}b''(\eta_{\tilde{u}})A(u-\tilde{u})\Big\}$$

where $\eta_u = \beta + Au$, $\eta_{\tilde{u}} = \beta + A\tilde{u}$ and $\tilde{u}$ is the maximizing value of $G(u)$.

Numerical integration of $\mathcal{E}\{s(u)\}$ was carried out using 9 node Gaussian quadrature over a $7 \times 8 \times 8$ grid of the parameter space for $(\beta, \sigma_1^2, \sigma_2^2)$, with each range of values encompassing the MLE (-0.75, 0.20, 0.01), obtained from both Laplace and 9 node Gaussian quadrature. The seven grid points taken for β were computed as -0.75 , $\pm$ 1se, $\pm$ 2se, and $\pm$ 3se (using the REML estimated se) and encompasses the likelihood ratio 95% confidence interval for β, (-1.084, -0.424). The grid of σ_1^2 included the value 0, and the corresponding endpoints of 50-, 67-, 95- and 99-% confidence intervals, assuming that $\hat{\sigma}_1^2/(8\sigma_1^2) \sim \chi_8^2$. The grid for σ_2^2 was similarly defined, assuming $\hat{\sigma}_2^2/(26\sigma_2^2) \sim \chi_{26}^2$. A portion of the 448 evaluations of $ln(\mathcal{E}\{s(u)\})$ are given in Table 5.

The most obvious feature of the tabulated values is that for any fixed value of σ_2^2, the log relative error with $\sigma_1^2 > 0.04$ is basically constant; whereas for any β and σ_1^2, there is an increase in log relative error as σ_2^2 increases. This suggests that the gradient and Hessian matrix for $\mathcal{E}\{s(u)\}$ are composed of 0's (or approximately so) except for the two terms corresponding to gradient and curvature along σ_2^2. It also suggests that a profile likelihood plot for σ_2^2 based on the Laplace approximation will be exact. As β increases towards 0, there is a slight decrease in the relative error, with the changes more prominent as σ_2^2 increases. The tabulated values suggest that the maximum relative error in the Laplace approximation is about 1.05%, and fairly constant over a reasonable range of the parameter space. Hence, the MLE and profile likelihood for all parameters should be very close to exact, as evidenced by the log relative profile likelihood plot in Figure 1 (see Section 1.4.1).

The point of the tabulation of $\mathcal{E}\{s(u)\}$ is seen by considering a likelihood ratio test for H_0: $\beta = \beta_0$. The likelihood ratio test statistic based on the Laplace approximation is

$$\lambda_{La} = 2\{l_{La}(\hat{\beta}, \hat{\sigma}_1^2, \hat{\sigma}_2^2) - l_{La}(\beta_0, \hat{\sigma}_{1\beta_0}^2, \hat{\sigma}_{2\beta_0}^2)\}$$

Table 5. Log Relative Error of Laplace Approximation - Cell Irradiation

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0046	0.0058	0.0096	0.0148	0.0173
	0	0	0.00070	0.00110	0.00257	0.00459	0.00550
	.0672	0.00207	0.00479	0.00544	0.00721	0.00910	0.00987
-1.2318	.1808	0.00228	0.00511	0.00578	0.00760	0.00953	0.01032
	.3817	0.00235	0.00522	0.00589	0.00773	0.00969	0.01048
	0	0	0.00087	0.00130	0.00281	0.00480	0.00569
	.0672	0.00206	0.00479	0.00543	0.00720	0.00909	0.00986
-1.0722	.1808	0.00227	0.00511	0.00577	0.00759	0.00953	0.01031
	.3817	0.00235	0.00522	0.00589	0.00773	0.00969	0.01048
	0	0	0.00118	0.00166	0.00322	0.00515	0.00599
	.0672	0.00205	0.00479	0.00543	0.00720	0.00908	0.00985
-0.7532	.1808	0.00227	0.00511	0.00577	0.00759	0.00952	0.01030
	.3817	0.00234	0.00522	0.00589	0.00773	0.00968	0.01047
	0	0	0.00142	0.00195	0.00354	0.00542	0.00623
	.0672	0.00204	0.00479	0.00543	0.00720	0.00907	0.00983
-0.4342	.1808	0.00226	0.00510	0.00577	0.00758	0.00951	0.01029
	.3817	0.00234	0.00522	0.00589	0.00773	0.00968	0.01047
	0	0	0.00152	0.00205	0.00366	0.00553	0.00633
	.0672	0.00203	0.00478	0.00543	0.00719	0.00907	0.00982
-0.2746	.1808	0.00226	0.00510	0.00577	0.00758	0.00951	0.01029
	.3817	0.00234	0.00521	0.00589	0.00772	0.00967	0.01046

The five β-values correspond to endpoints of 50-, 95- and 99-% confidence intervals.
The three non-zero σ_1^2-values correspond to 50- and 95-% confidence interval endpoints.
The 5 non-zero σ_2^2-values correspond to 50-, 95- and 99-% confidence interval endpoints.

$$= 2\{l(\hat{\beta}, \hat{\sigma}_1^2, \hat{\sigma}_2^2) + error_1 - l(\beta_0, \hat{\sigma}^2_{1\beta_0}, \hat{\sigma}^2_{2\beta_0}) - error_2\}$$

$$= \lambda_{exact} + 2(error_1 - error_2)$$

(ignoring differences in MLEs). For the range of parameters considered here, the maximum possible value of $2(error_1 - error_2)$ is $2(.01048 - .00070) = .01956$, which is a tiny error in a statistic that is to be compared to a χ_1^2 distribution.

3.3.2.2 Relative Error for Simulated Situations

To further explore the error in the Laplace approximate likelihood and its effect on inferences, seven data sets were generated using the same experimental setup from the Cell Irradiation study, with true parameters altered somewhat. The assumed model for which the data were generated was of the form:

$$Y|U \sim binomial(m, \pi); \quad logit(\pi) = \beta_0 + \beta_1 X + AU; \quad X_i \sim Uniform(0, 1);$$

$$U \sim MVN(0, D) \; ; \; D = \begin{bmatrix} \sigma_1^2 I_1 & 0 \\ 0 & \sigma_2^2 I_2 \end{bmatrix} .$$

Seven data sets were generated with the following parameter settings:

Data Set #	β_0	β_1	σ_1^2	σ_2^2	m
1	1	0	.2	.1	25
2	1	0	.2	.1	5
3	1	0	.1	.2	5
4	1	0	.4	.2	25
5	0	3	.2	.1	2
6	0	3	.2	.1	5
7	0	3	.2	.1	10

For each generated data set, numerical integration of $\mathfrak{E}\{s(u)\}$ was carried out using 9 node Gaussian quadrature over a $7 \times 8 \times 8$ grid of the parameter space: (β_0, σ_1^2, σ_2^2) for data sets 1-4, and (β_1, σ_1^2, σ_2^2) for data sets 5-7. The grid points were computed in the same manner as those for the Cell Irradiation data reported in Table 5, with true parameter values used in place of estimates where

appropriate. The results are given in Tables 6-12, with Table 13 summarizing information across the data sets.

For both the simulated and the Cell Irradiation data, maximum values for *Log Relative Error* occurred at the maximum tabulated (σ_1^2, σ_2^2) within the minimum tabulated β-value. Additionally, the magnitude of *Log Relative Error* appears to be directly related to the range of σ_2^2—for values of σ_2^2 around 0.01 to 0.025, tabled entries are similar in magnitude for all data sets. This raises the question: if one used the same absolute grid for the dispersion components in evaluating $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$ for all data sets, would one get similar values (in magnitude)? From Table 13, there is no strong support that the error in λ is related to the binomial index, m. Data sets 5, 6, and 7 have very similar maximum errors, which are lower than those for sets 1–4. However, within the two types of studies they represent, there seems to be no evidence of differences.

From the simulation results and from the Cell Irradiation data, it is strongly suggested that the shape of the log Laplace approximate likelihood is not practically different from the shape of the log-likelihood— the two differ mainly in a shift up or down. Hence, the parameters that maximize the Laplace approximate likelihood also maximize the Likelihood function. Additionally, maximizing profile log Laplace maximizes the profile log-likelihood. These imply that the Laplace method leads to nearly exact MLEs and LR inferences.

Table 6. Log Relative Error of Laplace Approximation - Simulated Data Set 1

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0469	0.0592	0.0974	0.1496	0.1755
	0	0	0.01159	0.01627	0.03155	0.05151	0.06078
	.0683	0.00802	0.02673	0.03192	0.04780	0.06774	0.07689
0.3411	.1835	0.01764	0.03891	0.04456	0.06158	0.08252	0.09201
	.3875	0.02650	0.04955	0.05560	0.07373	0.09590	0.10588
	0	0	0.01000	0.01423	0.02847	0.04783	0.05703
	.0683	0.00761	0.02517	0.03015	0.04558	0.06529	0.07443
0.5607	.1835	0.01751	0.03810	0.04363	0.06039	0.08121	0.09069
	.3875	0.02663	0.04930	0.05528	0.07327	0.09538	0.10535
	0	0	0.00595	0.00901	0.02064	0.03851	0.04753
	.0683	0.00656	0.02147	0.02594	0.04028	0.05939	0.06848
1.0000	.1835	0.01715	0.03625	0.04150	0.05767	0.07817	0.08763
	.3875	0.02690	0.04874	0.05458	0.07225	0.09419	0.10416
	0	0	0.00157	0.00317	0.01125	0.02678	0.03538
	.0683	0.00520	0.01701	0.02086	0.03379	0.05203	0.06100
1.4393	.1835	0.01661	0.03405	0.03898	0.05443	0.07449	0.08390
	.3875	0.02715	0.04811	0.05377	0.07108	0.09281	0.10276
	0	0	-0.00041	0.00040	0.00633	0.02019	0.02843
	.0683	0.00439	0.01453	0.01802	0.03010	0.04777	0.05664
1.6589	.1835	0.01626	0.03280	0.03755	0.05259	0.07239	0.08175
	.3875	0.02726	0.04775	0.05333	0.07043	0.09203	0.10197

Maximum error in λ: $2(.10588 + .00041) = .21258$

Table 7. Log Relative Error of Laplace Approximation - Simulated Data Set 2

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0469	0.0592	0.0974	0.1496	0.1755
	0	0	0.00355	0.00545	0.01335	0.02757	0.03569
	.0683	0.00526	0.01513	0.01829	0.02933	0.04637	0.05545
0.3624	.1835	0.02093	0.03535	0.03945	0.05291	0.07228	0.08220
	.3875	0.04587	0.06363	0.06846	0.08388	0.10527	0.11598
	0	0	0.00292	0.00451	0.01122	0.02359	0.03077
	.0683	0.00455	0.01312	0.01590	0.02571	0.04110	0.04940
0.5749	.1835	0.01927	0.03239	0.03615	0.04859	0.06666	0.07599
	.3875	0.04422	0.06093	0.06550	0.08015	0.10060	0.11088
	0	0	0.00134	0.00214	0.00580	0.01337	0.01814
	.0683	0.00272	0.00805	0.00987	0.01659	0.02783	0.03418
1.0000	.1835	0.01500	0.02498	0.02792	0.03787	0.05281	0.06071
	.3875	0.04001	0.05432	0.05829	0.07116	0.08942	0.09872
	0	0	-0.00011	-0.00009	0.00044	0.00273	0.00468
	.0683	0.00075	0.00258	0.00334	0.00652	0.01281	0.01676
1.4251	.1835	0.00967	0.01604	0.01803	0.02502	0.03618	0.04233
	.3875	0.03444	0.04600	0.04927	0.06007	0.07578	0.08392
	0	0	-0.00065	-0.00093	-0.00172	-0.00187	-0.00130
	.0683	-0.00014	0.00008	0.00033	0.00175	0.00546	0.00810
1.6376	.1835	0.00675	0.01124	0.01272	0.01813	0.02720	0.03237
	.3875	0.03110	0.04119	0.04409	0.05375	0.06805	0.07555

Maximum error in λ: $2(.11598 + .00187) = .23570$

Table 8. Log Relative Error of Laplace Approximation - Simulated Data Set 3

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0938	0.1183	0.1949	0.2991	0.3511
	0	0	0.01341	0.01990	0.04414	0.08147	0.10052
	.0341	0.00177	0.02140	0.02865	0.05411	0.09162	0.11050
0.2040	.0917	0.00847	0.03373	0.04168	0.06827	0.10596	0.12467
	.1938	0.02182	0.05150	0.06005	0.08769	0.12569	0.14433
	0	0	0.01126	0.01678	0.03770	0.07069	0.08785
	.0341	0.00148	0.01809	0.02433	0.04657	0.08013	0.09732
0.4693	.0917	0.00730	0.02906	0.03604	0.05970	0.09395	0.11122
	.1938	0.01959	0.04587	0.05355	0.07869	0.11384	0.13129
	0	0	0.00498	0.00776	0.01963	0.04139	0.05381
	.0341	0.00065	0.00901	0.01255	0.02636	0.04987	0.06286
1.0000	.0917	0.00408	0.01692	0.02143	0.03771	0.06342	0.07713
	.1938	0.01370	0.03177	0.03738	0.05648	0.08476	0.09933
	0	0	-0.00100	-0.00106	0.00068	0.00869	0.01499
	.0341	-0.00014	-0.00017	0.00040	0.00444	0.01546	0.02301
1.5307	.0917	0.00054	0.00368	0.00535	0.01286	0.02793	0.03708
	.1838	0.00644	0.01530	0.01849	0.03043	0.05029	0.06127
	0	0	-0.00308	-0.00427	-0.00702	-0.00604	-0.00315
	.0341	-0.00041	-0.00375	-0.00449	-0.00514	-0.00079	0.00366
1.7960	.0917	-0.00101	-0.00226	-0.00198	0.00101	0.01019	0.01670
	.1938	0.00263	0.00689	0.00883	0.01694	0.03208	0.04100

Maximum error in λ: $2(.14433 + .00702) = .30270$

Table 9. Log Relative Error of Laplace Approximation - Simulated Data Set 4

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0938	0.1183	0.1949	0.2991	0.3511
	0	0	0.03194	0.04166	0.06888	0.09940	0.11257
	.1365	0.01439	0.05513	0.06498	0.09237	0.12281	0.13577
0.1326	.3670	0.02596	0.07006	0.08065	0.10997	0.14212	0.15561
	.7750	0.03503	0.08154	0.09274	0.12381	0.15782	0.17202
	0	0	0.02871	0.03794	0.06460	0.09547	0.10897
	.1365	0.01406	0.05322	0.06290	0.09015	0.12083	0.13399
0.4217	.3670	0.02609	0.06934	0.07984	0.10907	0.14136	0.15498
	.7750	0.03534	0.08141	0.09255	0.12358	0.15769	0.17197
	0	0	0.01930	0.02726	0.05264	0.08454	0.09894
	.1365	0.01315	0.04856	0.05782	0.08458	0.11571	0.12930
1.0000	.3670	0.02629	0.06768	0.07795	0.10693	0.13946	0.15331
	.7750	0.03598	0.08110	0.09214	0.12305	0.15733	0.17176
	0	0	0.00746	0.01337	0.03610	0.06880	0.08428
	.1365	0.01176	0.04264	0.05130	0.07725	0.10871	0.12274
1.5783	.3670	0.02640	0.06567	0.07567	0.10426	0.13695	0.15102
	.7750	0.03666	0.08075	0.09165	0.12241	0.15682	0.17139
	0	0	0.00150	0.00597	0.02638	0.05897	0.07494
	.1365	0.01085	0.03917	0.04745	0.07282	0.10435	0.11860
1.8674	.3670	0.02640	0.06452	0.07435	0.10270	0.13542	0.14960
	.7750	0.03700	0.08055	0.09138	0.12204	0.15650	0.17113

Maximum error in λ: $2(.17202 - .00150) = .34104$

Table 10. Log Relative Error of Laplace Approximation - Simulated Data Set 5

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0469	0.0592	0.0974	0.1496	0.1755
	0	0	0.00121	0.00189	0.00488	0.01075	0.01433
	.0683	0.00220	0.00618	0.00750	0.01227	0.02013	0.02454
0.9445	.1835	0.01165	0.01851	0.02049	0.02713	0.03709	0.04239
	.3875	0.03352	0.04298	0.04557	0.05392	0.06583	0.07195
	0	0	0.00061	0.00096	0.00254	0.00576	0.00778
	.0683	0.00120	0.00341	0.00416	0.00691	0.01156	0.01423
1.6294	.1835	0.00719	0.01143	0.01268	0.01689	0.02332	0.02680
	.3875	0.02345	0.02996	0.03175	0.03758	0.04598	0.05034
	0	0	0.00005	0.00008	0.00022	0.00055	0.00079
	.0683	0.00013	0.00039	0.00048	0.00085	0.00154	0.00197
3.0000	.1835	0.00117	0.00194	0.00218	0.00303	0.00442	0.00522
	.3875	0.00619	0.00807	0.00861	0.01040	0.01309	0.01454
	0	0	0.00006	0.00010	0.00023	0.00046	0.00058
	.0683	0.00011	0.00027	0.00033	0.00050	0.00074	0.00086
4.3704	.1835	0.00029	0.00045	0.00050	0.00064	0.00086	0.00097
	.3875	-0.00020	-0.00002	0.00004	0.00026	0.00063	0.00084
	0	0	0.00012	0.00018	0.00044	0.00091	0.00117
	.0683	0.00021	0.00054	0.00065	0.00100	0.00151	0.00177
5.0555	.1835	0.00075	0.00111	0.00121	0.00150	0.00190	0.00209
	.3875	-0.00002	0.00011	0.00014	0.00026	0.00045	0.00055

Maximum error in λ: $2(.07195 + .00020) = .14430$

Table 11. Log Relative Error of Laplace Approximation - Simulated Data Set 6

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0469	0.0592	0.0974	0.1496	0.1755
	0	0	0.00289	0.00442	0.01063	0.02149	0.02756
	.0683	0.00402	0.01116	0.01341	0.02118	0.03299	0.03921
0.8710	.1835	0.01476	0.02379	0.02636	0.03476	0.04687	0.05309
	.3875	0.03167	0.04123	0.04387	0.05239	0.06446	0.07060
	0	0	0.00149	0.00229	0.00566	0.01178	0.01532
	.0683	0.00226	0.00625	0.00754	0.01207	0.01918	0.02302
1.5806	.1835	0.00943	0.01489	0.01647	0.02173	0.02951	0.03359
	.3875	0.02292	0.02914	0.03088	0.03660	0.04488	0.04918
	0	0	0.00010	0.00015	0.00036	0.00073	0.00093
	.0683	0.00022	0.00057	0.00068	0.00105	0.00160	0.00190
3.0000	.1835	0.00125	0.00176	0.00192	0.00246	0.00333	0.00383
	.3875	0.00543	0.00625	0.00651	0.00744	0.00898	0.00986
	0	0	0.00014	0.00021	0.00040	0.00049	0.00040
	.0683	0.00021	0.00043	0.00047	0.00048	0.00016	-0.00016
4.4194	.1835	-0.00020	-0.00052	-0.00065	-0.00117	-0.00220	-0.00284
	.3875	-0.00292	-0.00404	-0.00436	-0.00540	-0.00694	-0.00775
	0	0	0.00028	0.00042	0.00092	0.00157	0.00180
	.0683	0.00045	0.00107	0.00123	0.00164	0.00192	0.00190
5.1290	.1835	0.00070	0.00083	0.00081	0.00063	0.00001	-0.00045
	.3875	-0.00272	-0.00372	-0.00402	-0.00507	-0.00673	-0.00765

Maximum error in λ: $2(.07060 + .00775) = .15670$

Table 12. Log Relative Error of Laplace Approximation - Simulated Data Set 7

Entries are $l(\beta,\sigma_1^2,\sigma_2^2) - l_{La}(\beta,\sigma_1^2,\sigma_2^2)$

		σ_2^2					
β	σ_1^2	0	0.0469	0.0592	0.0974	0.1496	0.1755
	0	0	0.00208	0.00320	0.00783	0.01618	0.02098
	.0683	0.00346	0.00974	0.01176	0.01878	0.02957	0.03531
2.0015	.1835	0.01471	0.02432	0.02708	0.03614	0.04912	0.05574
	.3875	0.03293	0.04521	0.04859	0.05941	0.07440	0.08188
	0	0	0.00140	0.00218	0.00551	0.01182	0.01557
	.0683	0.00275	0.00765	0.00925	0.01491	0.02382	0.02866
2.3343	.1835	0.01319	0.02135	0.02371	0.03153	0.04289	0.04876
	.3875	0.03134	0.04227	0.04530	0.05504	0.06865	0.07550
	0	0	0.00054	0.00086	0.00233	0.00544	0.00744
	.0683	0.00165	0.00440	0.00532	0.00870	0.01430	0.01746
3.0000	.1835	0.01032	0.01593	0.01758	0.02314	0.03145	0.03583
	.3875	0.02801	0.03642	0.03878	0.04645	0.05737	0.06294
	0	0	0.00015	0.00025	0.00076	0.00198	0.00285
	.0683	0.00097	0.00239	0.00287	0.00470	0.00786	0.00972
3.6657	.1835	0.00788	0.01153	0.01262	0.01636	0.02209	0.02519
	.3875	0.02466	0.03092	0.03269	0.03854	0.04703	0.05142
	0	0	0.00007	0.00012	0.00038	0.00106	0.00157
	.0683	0.00076	0.00177	0.00212	0.00342	0.00572	0.00710
3.9985	.1835	0.00687	0.00977	0.01064	0.01365	0.01833	0.02089
	.3875	0.02306	0.02838	0.02990	0.03494	0.04236	0.04623

Maximum error in λ: $2(.08188 - .00007) = .16362$

Table 13. Summary of Log Relative Errors of Laplace Approximation

Data Set #	β_0	β_1	σ_1^2	σ_2^2	m	Maximum Error in λ
1	1	0	.2	.1	25	.21258
2	1	0	.2	.1	5	.23570
3	1	0	.1	.2	5	.30270
4	1	0	.4	.2	25	.34104
5	0	3	.2	.1	2	.14430
6	0	3	.2	.1	5	.15670
7	0	3	.2	.1	10	.16362
Cell Irradiation	-.75*	0	.2*	.01*	400	.01956

*: *Laplace estimated parameter values.*

4. COMPARISON OF ANALYSIS RESULTS

If one believes that likelihood analysis is the best approach to a generalized linear model with several random effects, the results of the previous chapter suggest that it may be carried out with the Laplace approximation. In many problems one must weigh the tradeoff between the desirable likelihood ratio inferences and those from the more robust methods based on moment assumptions. Although robustness is not explored in this thesis, one issue that is relevant is whether the moment-based methods provide estimates and inferences that are nearly as good as those based on likelihood theory when, in fact, the presumed model is correct. This is explored herein by numerical comparisons of the Laplace-likelihood procedure with quasi-likelihood (QL), iterative generalized least squares (IGLS), and REML-like (REML) procedures. The three example data sets from Chapter 1 and simulated samples based on the Cell Irradiation design are used to compare estimators and inferences for single coefficients.

Because maximum likelihood and iterative weighted least squares estimators are identical for generalized linear models, it is suspected that in this extension of generalized linear models the maximum likelihood estimators and those based on iterative generalized least squares or iteratively weighted least squares (such as REML) may also be quite similar. A more important question is the comparison of tests and confidence intervals. As long as there are not too many nuisance parameters, likelihood ratio inferences are expected to be quite good. This is true whether or not the sampling distribution of the estimator is approximately normal. The inferences based on the moment-based methods, however, are adequate according to the approximate normality of the estimators. Therefore all methods should give reasonably good inferences when sample sizes are large, binomial indexes are large, or Poisson means are large. However, it is suspected that likelihood-ratio inferences are better more generally.

4.1 Comparison of Resulting Analyses from Example Data Sets

For each of the three data sets described in Chapter 1 (Cell Irradiation, Revertant Colony, Salamander Mating), analyses were carried out by each of QL

(§2.2.5), IGLS (§2.3.2), REML (§2.3.3), and Maximum Likelihood via Laplace approximate likelihood (§3.2). Additionally for the Cell Irradiation data, Maximum Likelihood via numerically integrated likelihood (§2.3.1, 3.3.2) utilizing 2-, 5- and 9-node Gauss-Hermite quadrature was done. For moment based methods, inferences were carried out using the asymptotic results given in §2.4. The Nelder-Mead simplex algorithm was employed for likelihood optimization required in the Maximum Likelihood analyses, with inferences drawn by following the likelihood ratio ideas in §3.2.3.

The results of the analyses are presented in the following sub-sections. As expected, the computing times for estimation of the Laplace approximate MLEs was substantially longer than times for the moments-based methods, due mainly to the slow convergence rate of the Nelder-Mead algorithm. However, MLEs and Likelihood Ratio inferences were obtained much faster than from numerical integration, and obtained for data sets where numerical integration was infeasible (i.e., Revertant Colony and Salamander Mating experiments).

A recurring aspect in all analyses here was that the MLEs for the β-coefficients were always the most "extreme" (i.e., largest in magnitude). The MLEs were always furthest from 0, followed by the REML and then IGLS estimates in order of magnitude. For the binomial data sets, QL and IGLS estimates were identical, as mentioned in McCullagh and Nelder (1989), suggesting that for balanced designs IGLS estimates of β are unaffected by how (or even if) the dispersion components are estimated.

From the profile likelihood plots (Figs. 1, 3, 4, 5) it is very apparent that poor inferences can result when using QL, with results being either very conservative (SE too large and thus fail to reject too often) or liberal (SE too small and reject too often).[5] The REML profile likelihood plots, on the whole, follow those of Laplace more so than of the IGLS plots, due partially to the shifting away from 0 seen in Laplace and REML estimates. In most instances, the REML confidence interval endpoints were about mid-way between respective endpoints from IGLS and Laplace. For the Cell Irradiation data, the Laplace profile plot is indistinguishable from numerical integration, and is not far from being quadratic—due possibly to observations being binomial counts out of 400.

[5]Recall that this is similar to problems with the t-test when correlations between observations are ignored.

For the Salamander Mating data, the Laplace plots have pronounced skewness, with heavier "tails" away from 0 (i.e., toward more extreme endpoints)—due most likely to the data being Bernoulli observations.

As for the dispersion components estimates among the methods, all seemed to give fairly similar estimates, especially in the Cell Irradiation and Revertant Colony data sets. The REML dispersion component estimates were always the larger values due to use of shrinkage estimates.

4.1.1 Comparison of Analyses of Cell Irradiation Data

The Cell Irradiation data introduced in Section 1.4.1 was modeled assuming that conditional on the trial and dish effects being known, the number of surviving cells in each dish is distributed as binomial with mean $400\pi_{ij}$ ($i = 1,\ldots,9;\ j = 1,2,3$) and that

$$ln[\pi_{ij}/(1-\pi_{ij})] = \beta + u_{1i} + u_{2ij},$$

where β is the natural log-odds of survival, u_1 represents the 9 random effects due to trials and u_2 represents the 27 random effects due to each dish. It will be further assumed that $u_1 \sim N_9(\mathbf{0}, \sigma_1^2\boldsymbol{I})$, $u_2 \sim N_{27}(\mathbf{0}, \sigma_2^2\boldsymbol{I})$ and u_1 and u_2 are independent. The resulting estimates and inferences are given in Table 14.[6]

The tabled results suggest that maximizing the Laplace approximate likelihood lessened computing time but gave up nothing in estimation and inferences as compared to numerical integration of the likelihood function. This is in complete agreement with the results of §3.3.2.1, where the relative error in Laplace was examined for this data and found to be constant.

Displayed in Figure 1 (given in §1.4.1) are log profile likelihood and pseudo log profile likelihood plots for each of the methods.[7] The two indistinguishable curves for Laplace and Maximum Likelihood via numerical

[6]The actual implementation of Gaussian quadrature to evaluate the likelihood function at a point in the parameter space computed the Laplace approximate likelihood at that point, and then used Gaussian quadrature (2–, 5–, and 9–node) to evaluate the error in the Laplace approximation, $\mathcal{E}\{s(u)\}$, as discussed in §3.3.2 .

[7]The lack of smoothness in the curves is an artifact of using a limited number of β values on which to evaluate each curve. No attempt of smoothing the curves has been taken.

Table 14. Estimates and Confidence Intervals for Cell Irradiation Data

	Method						
					Gaussian Quadrature		
	QL	IGLS	REML	Laplace	2-node	5-node	9-node
Estimate							
$\hat{\beta}$	-.718673	-.718673	-.751901	-.753196	-.753191	-.753196	-.753178
$\hat{\sigma}_1^2$	NA	.185889	.221660	.196993	.196986	.196983	.196994
$\hat{\sigma}_2^2$	NA	.009952	.009838	.009868	.009883	.009890	.009890
$\hat{\phi}$	18.9603	NA	NA	NA	NA	NA	NA
$\text{Var}(\hat{\beta})$	.007969	.021443	.025443	NA	NA	NA	NA
Time to Compute Estimates (in seconds):							
	.5318	1.1389	2.1412	51.7145	152.1142	3119.4092	31883.9002
95% Confidence Interval for β:							
Lower	-.89364	-1.00569	-1.06459	-1.08249	-1.08280	-1.08251	-1.08251
Upper	-.54371	-.43166	-.43931	-.42487	-.42487	-.42487	-.42487
Estimates and 95% Confidence Intervals for $\pi^{\dagger}$:							
Lower	.29036	.26783	.25643	.25303	.25298	.25303	.25303
$\hat{\pi}$	.32769	.32769	.32041	.32013	.32013	.32013	.32013
Upper	.36733	.39365	.39190	.39535	.39535	.39535	.39535

†: $\pi = logit^{-1}(\beta)$
NA: Not Applicable

integration are their respective log profile likelihood curves (see §3.2.3). These are constructed by plotting

$$\ell(\beta, \tilde{\sigma}^2_{1,\beta}, \tilde{\sigma}^2_{2,\beta}) - \ell(\hat{\beta}, \hat{\sigma}^2_1, \hat{\sigma}^2_2) \text{ versus } \beta$$

(i.e., $-\frac{1}{2}\lambda_{(\beta)}$ vs. β, where $\lambda_{(\beta)}$ is the LR test statistic for a two-sided test with

null hypothesis value β). The 3 curves plotted for QL, IGLS and REML are "pseudo" log profile likelihood plots, constructed by plotting

$$-\tfrac{1}{2}z^2_{(\beta)} \text{ vs. } \beta$$

where $z_{(\beta)}$ is their respective z-statistic for the same two-sided test. The horizontal reference line is at -1.92 (i.e., $-\frac{1}{2}\chi^2_1$). The rationale for using such a plot is from noting that under the null hypothesis, $\lambda_{(\beta)}$ and $z^2_{(\beta)}$ are both approximately distributed χ^2_1, thus allowing a visual comparison of each method's $(1-\alpha)$-level confidence interval or significance levels for various null hypotheses. In particular, the plots show that QL, IGLS and REML significance levels for nearly any null hypothesized β value will be lower than those from Maximum Likelihood, as their curves are contained within the log relative profile curve. Also, any level $(1-\alpha)$ confidence interval differs only in the lower endpoint amongst the methods (except for QL where both endpoints differ from the other methods).

When looking at the confidence intervals for β (and subsequently for π), there is close agreement between REML and Maximum Likelihood.[8] This is not unexpected, however, as the sample proportions are in the range of .2 to .8 and the counts are out of 400 cells. One would feel that the central limit theorem would be helping REML to do a fairly good job in estimation and inference.

4.1.2 Comparison of Analyses of Revertant Colony Data

The analyses of the quadratic dose-response model, introduced in Section 1.4.2, are summarized in Table 15. This quadratic dose-response model assumed that conditional on the replicate and plate effects being known, the number of revertant colonies on each plate is distributed as Poisson with mean μ_{ijk} ($i = 1, 2, 3$; $j = 1, 2, 3$; $k = 1, \ldots, 6$) and that

$$ln(\mu_{ijk}) = \beta_0 + \beta_1 D_{ijk} + \beta_2 D^2_{ijk} + \beta_3 Z_{ijk} + u_{1i} + u_{2jk(i)}$$

where $D = ln(\text{Dose} + 1)$, $Z = \begin{cases} 1 & \text{if dose } = 0 \\ 0 & \text{otherwise} \end{cases}$, u_1 represents the 3 random

[8] Notice similarity of REML and Maximum Likelihood curves in Figure 1.

Table 15. Estimates and Confidence Intervals for Revertant Colony Data

	Method			
	QL	IGLS	REML	Laplace
Estimate				
$\hat{\beta}_0$	-.565596	-.507555	-.618757	-.638892
$\hat{\beta}_1$	1.447075	1.430449	1.438344	1.443590
$\hat{\beta}_2$	-.117916	-.116747	-.116676	-.117134
$\hat{\beta}_3$	3.620644	3.562603	3.643270	3.653363
$\hat{\sigma}_1^2$	NA	.091919	.1129936	.075284
$\hat{\sigma}_2^2$	NA	.053573	.035158	.031753
$\hat{\phi}$	5.594421	NA	NA	NA
$\text{Var}(\hat{\beta}_0)$	1.6684	.86081	.69936	NA
$\text{Var}(\hat{\beta}_1)$	.15558	.07612	.06081	NA
$\text{Var}(\hat{\beta}_2)$	.000848	.000403	.000323	NA
$\text{Var}(\hat{\beta}_3)$	1.6977	.841361	.670826	NA
Time to Compute Estimates (in seconds):				
	.8924	2.6540	5.8659	836.5971
Estimates and 95% Confidence Intervals for γ:				
Lower	5.6831†	5.7877†	5.8729†	5.7920
$\hat{\gamma} = -\frac{\hat{\beta}_1}{2\hat{\beta}_2}$	6.13604	6.12419	6.16306	6.16209
Upper	6.5890†	6.4607†	6.4549†	6.4046

†: Standard Errors Calculated via Delta-Method (Mood, Graybill and Boes, 1974 page 181)
NA: Not Applicable

effects due to replicate and u_2 represents the 54 random effects due to each plate. It was further assumed that there is no dose–by–replicate interaction, that $u_1 \sim N_3(\mathbf{0}, \sigma_1^2 \boldsymbol{I})$, $u_2 \sim N_{54}(\mathbf{0}, \sigma_2^2 \boldsymbol{I})$, and that u_1 and u_2 are independent.

All methods yield very similar estimates for β_1, β_2, β_3 and differ really only in their estimation of β_0. Again, the Laplace estimate of β_0 is the most extreme (further from 0), with REML nearly as extreme, and QL and IGLS least extreme. The confidence intervals for γ ($= ln[\text{Dose}_{Max} + 1]$) differ little except for QL, due to the much larger variance estimates for the β parameters. Figure 3 (given in §1.4.2) presents log profile likelihood plots for $e^{\gamma} - 1$, as well as pseudo plots for QL, IGLS and REML results.[9] The figure suggests p-values will be larger for the methods as compared to Laplace, and thus wider confidence intervals. This is especially true for QL results. Due to the nature of the data, one would not expect much disagreement between the methods, since the majority of the counts are greater than 20.

4.1.3 Comparison of Analyses of the Salamander Mating Data Sets

The data for the first Salamander Mating experiment was given in Section 1.4.3, along with the parametrization that assumed that conditional on the male and female effects being known, the mating failure/success outcome between Female i and Male j is a Bernoulli random variable with probability of success denoted by π_{ij} $(i = 1,\ldots,20;\ j = 1,\ldots,20)$ and that

$$ln[\pi_{ij}/(1-\pi_{ij})] = \beta_0 + \beta_1 F_i + \beta_2 M_j + \beta_3 FM_{ij} + u_{Fi} + u_{Mj} \quad ,$$

where

$$F_i = \begin{cases} 1 & \text{if Female } i \text{ is white-side} \\ 0 & \text{otherwise} \end{cases} \quad ,$$

$$M_j = \begin{cases} 1 & \text{if Male } j \text{ is white-side} \\ 0 & \text{otherwise} \end{cases} \quad ,$$

$$FM_{ij} = \begin{cases} 1 & \text{if Female } i \text{ and Male } j \text{ are white-side} \\ 0 & \text{otherwise} \end{cases} \quad ,$$

u_{Fi} represents the 20 female random effects and u_{Mj} represents the 20 male random effects. It was assumed that $u_F \sim N_{20}(\mathbf{0}, \sigma_F^2 \boldsymbol{I})$, $u_M \sim N_{20}(\mathbf{0}, \sigma_M^2 \boldsymbol{I})$ and u_F and u_M are independent.

[9] Although this data set has nested random effects, maximum likelihood via numerical integration was not done since there are 18 levels of the second random effect nested within each level of the first, requiring integrals over $\mathbb{R}^{19}$.

The analyses were carried out on each of three experiments, separately, with results given in Tables 16, 17 and 18.[10] Included in each table are results from applying the Gibbs Sampler (Zeger and Karim, 1991) technique, obtained from Karim and Zeger (1992).[11] Figures 4 and 5 present the log profile likelihood plots for the three experiments (Figure 4 is given in §1.4.3).

The computing times for all methods increased dramatically, reflecting the increased complexity in this data, as compared to the Cell Irradiation data. The computing times for Gibbs Sampler were not reported in Karim and Zeger (1992), although they reported times to be substantial. The curves for β_2 are quite similar in both shape and location. However, those for the other β parameters agree amongst the methods near 0, and differ with Laplace having heavier tails away from 0. Interestingly for QL and IGLS, there is little difference between the curves and inferences, irrespective of the parameter.

The methods all give rather different estimates for the two dispersion components, but with similarity in their relational aspects ($\hat{\sigma}_F^2 >> \hat{\sigma}_M^2$ for experiments 1 & 2, and $\hat{\sigma}_F^2 << \hat{\sigma}_M^2$).[12] An interesting result is that $\hat{\phi}$ is not really different from 1, is identical in all three experiments, and suggests no overdispersion in the ordinary GLM.

Gibbs Sampler estimates are even more extreme (i.e., further from 0) than those of Laplace in experiment 1, and similar to Laplace estimates in the other two experiments. The Gibbs Sampler intervals, which "correspond" to 90% confidence intervals, are shifted even further from 0 and would be wider than REML and Laplace intervals had they been 95% rather than 90%. Gibbs Sampler percentile endpoints suggest asymmetry as do the Laplace plots.

---

[10]The data for experiments 2 & 3 can be found in Tables 14.5 and 14.6 of McCullagh and Nelder, 1989.

[11]These have been included as an alternative modeling approach that has been given some attention recently, but was not part of the discussion of Chapter 2. Gibbs Sampler is used in conjunction with a Bayesian modeling approach to the data.

[12]The IGLS results in McCullagh and Nelder (1989) have identical $\hat{\beta}$ values as those given here, but have different dispersion components estimates (e.g., for experiment 1, $\hat{\sigma}_F^2 = 1.3704$ and $\hat{\sigma}_M^2 = .6963$: taken from Table 14.10, McCullagh and Nelder, 1989). An explanation for this is that McCullagh and Nelder develop quadratic forms from some differing approach than given in Chapter 2.

Table 16. Estimates and Confidence Intervals for First of Three Salamander Mating Experiments

	Method				
	QL	IGLS	REML	Laplace	Gibbs Sampler†
Estimate					
$\hat{\beta}_0$	1.011601	1.011601	1.164003	1.335250	1.48
$\hat{\beta}_1$	-2.20118	-2.20119	-2.57564	-2.94038	-3.25
$\hat{\beta}_2$	-.318453	-.318453	-.380461	-.422120	-.50
$\hat{\beta}_3$	2.355332	2.355336	2.806365	3.181243	3.62
$\hat{\sigma}^2_F$	NA	.879448	1.399325	1.574893	2.35
$\hat{\sigma}^2_M$	NA	.247360	.1053918	.0721079	.14
$\hat{\phi}$	1.034482	NA	NA	NA	NA
$\text{Var}(\hat{\beta}_0)$	.17633	.28314	.35599	NA	NA
$\text{Var}(\hat{\beta}_1)$	.36909	.53268	.71349	NA	NA
$\text{Var}(\hat{\beta}_2)$	.33151	.36993	.39625	NA	NA
$\text{Var}(\hat{\beta}_3)$	.68847	.66552	.81885	NA	NA
Time to Compute Estimates (in seconds):					
	1.5077	26.4631	92.2697	1834.6154	NR
95% Confidence Intervals for β_j:					
β_0 Lower	.188559	-.031324	-.005436	.120369	.30‡
Upper	1.83464	2.05453	2.33344	2.702903	2.91‡
β_1 Lower	-3.39194	-3.63170	-4.23121	-5.114146	-5.51‡
Upper	-1.01042	-.770681	-.920061	-1.288929	-1.59‡
β_2 Lower	-1.44695	-1.51056	-1.61425	-1.767010	-1.84‡
Upper	.810045	.873649	.853330	.849032	.65‡
β_3 Lower	.729041	.756379	1.03276	1.429819	1.76‡
Upper	3.98162	3.95429	4.57997	5.420920	5.85‡

†: From Table 3 of Karim and Zeger, 1992.
‡: 5^{th} and 95^{th} percentiles from estimated posterior distributions.
NA: Not Applicable; NR: Not Reported.

Table 17. Estimates and Confidence Intervals for Second Salamander Mating Experiment

	Method				
	QL	IGLS	REML	Laplace	Gibbs Sampler†
Estimate					
$\hat{\beta}_0$	.405465	.405465	.447181	.574436	.57
$\hat{\beta}_1$	-1.59505	-1.59505	-1.939048	-2.46328	-2.77
$\hat{\beta}_2$	-.538997	-.538997	-.595662	-.774200	-.75
$\hat{\beta}_3$	2.42172	2.42173	2.912344	3.70946	4.09
$\hat{\sigma}^2_F$	NA	.598820	1.247884	1.812176	2.99
$\hat{\sigma}^2_M$	NA	.240175	.604574	.917240	1.42
$\hat{\phi}$	1.034482	NA	NA	NA	NA
$\text{Var}(\hat{\beta}_0)$	.14368	.22279	.34308	NA	NA
$\text{Var}(\hat{\beta}_1)$	.33644	.44499	.66631	NA	NA
$\text{Var}(\hat{\beta}_2)$	.28222	.32085	.42909	NA	NA
$\text{Var}(\hat{\beta}_3)$	.63016	.60915	.76746	NA	NA
Time to Compute Estimates (in seconds):					
	1.17817	26.4707	100.7539	5151.7515	NR
95% Confidence Intervals for β_j:					
β_0 Lower	-.337471	-.519664	-.700854	-.849622	-.83‡
Upper	1.14840	1.33059	1.595217	2.126249	2.03‡
β_1 Lower	-2.73191	-2.90252	-3.538953	-4.872749	-5.04‡
Upper	-.453390	-.287583	-.339143	-.681095	-.93‡
β_2 Lower	-1.58024	-1.64922	-1.879556	-2.467011	-2.27‡
Upper	.502250	.571223	.688232	.752541	.67‡
β_3 Lower	.865827	.891982	1.195294	1.791425	2.23‡
Upper	3.97762	3.95147	4.629394	6.242727	6.51‡

†: From Table 3 of Karim and Zeger, 1992.
‡: 5^{th} and 95^{th} percentiles from estimated posterior distributions.
NA: Not Applicable; NR: Not Reported.

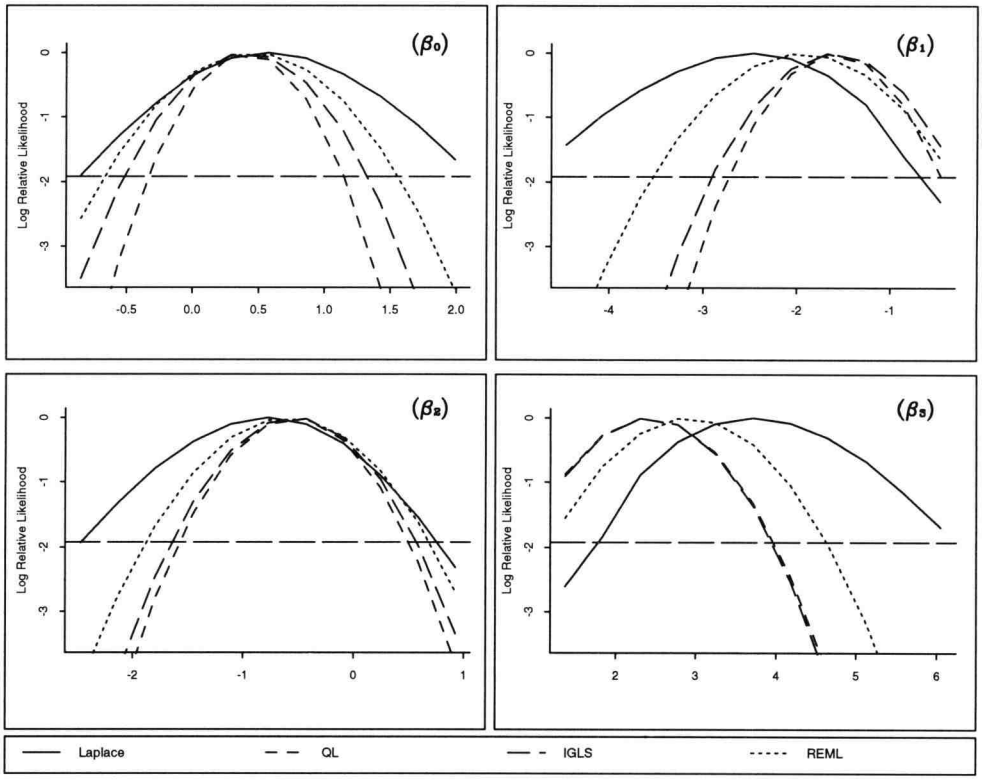
Table 18. Estimates and Confidence Intervals for Third Salamander Mating Experiment

	Method				
	QL	IGLS	REML	Laplace	Gibbs Sampler†
Estimate					
$\hat{\beta}_0$	.693147	.693147	.838298	1.016746	1.07
$\hat{\beta}_1$	-2.30242	-2.30259	-2.70322	-3.22504	-3.53
$\hat{\beta}_2$	-.559616	-.559616	-.677462	-.817223	-.81
$\hat{\beta}_3$	2.715597	2.715597	3.198355	3.820803	4.20
$\hat{\sigma}_F^2$	NA	.015237	.231860	.346552	.33
$\hat{\sigma}_M^2$	NA	.910365	1.497155	1.853219	2.89
$\hat{\phi}$	1.034455	NA	NA	NA	NA
$\text{Var}(\hat{\beta}_0)$	.15517	.24256	.35196	NA	NA
$\text{Var}(\hat{\beta}_1)$	.40341	.39305	.54754	NA	NA
$\text{Var}(\hat{\beta}_2)$	.29371	.46600	.64645	NA	NA
$\text{Var}(\hat{\beta}_3)$	.69044	.66747	.84211	NA	NA
Time to Compute Estimates (in seconds):					
	1.1744	27.5642	233.4699	12789.5969	NR
95% Confidence Intervals for β_j:					
β_0 Lower	-.078924	-.272161	-.324503	-.274582	-.12‡
Upper	1.465219	1.658455	2.001099	2.586007	2.48‡
β_1 Lower	-3.547312	-3.531377	-4.153537	-5.432116	-5.45‡
Upper	-1.057538	-1.073793	-1.252901	-1.662304	-2.07‡
β_2 Lower	-1.621841	-1.897596	-2.253342	-2.787078	-2.62‡
Upper	.5026090	.778365	.898418	.985311	.94‡
β_3 Lower	1.086819	1.114301	1.399726	1.954400	2.34‡
Upper	4.344055	4.316894	4.996984	6.251140	6.34‡

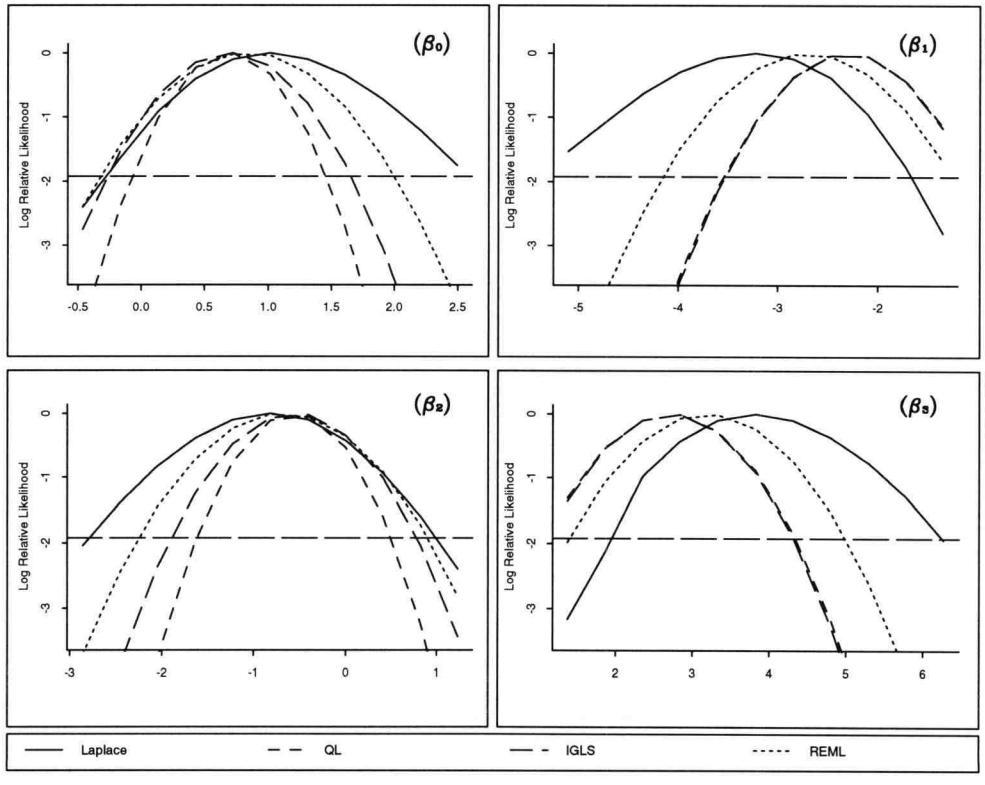
†: From Table 3 of Karim and Zeger, 1992.
‡: 5^{th} and 95^{th} percentiles from estimated posterior distributions.
NA: Not Applicable; NR: Not Reported.

Figure 5. Log Relative Profile Likelihood Plots - Salamander Mating: (a) Experiment 2; (b) Experiment 3

a)



b)



Estimates for probability of successful mating can be computed from the β estimates using the identities

$$\hat{\pi}_{RR} = logit^{-1}(\hat{\beta}_0) \ , \qquad \hat{\pi}_{RW} = logit^{-1}(\hat{\beta}_0 + \hat{\beta}_2) \ ,$$

$$\hat{\pi}_{WR} = logit^{-1}(\hat{\beta}_0 + \hat{\beta}_1) \ , \qquad \hat{\pi}_{WW} = logit^{-1}(\hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 + \hat{\beta}_3) \ .$$

Computed probabilities for the three experiments are given in Table 19. REML estimates are more extreme (i.e., probability estimates further from .5) than QL/IGLS, followed by Laplace and then Gibbs Sampler in "extremeness". The

Table 19. Successful Mating Probability Estimates for all Mating Experiments

	Method				
	QL	IGLS	REML	Laplace	Gibbs Sampler
Estimates for Experiment 1					
$\hat{\pi}_{RR}$	.733333	.733333	.762059	.791708	.814573
$\hat{\pi}_{RW}$	.666667	.666667	.686443	.713640	.727108
$\hat{\pi}_{WR}$	.233333	.233333	.195976	.167266	.145542
$\hat{\pi}_{WW}$	.700000	.700000	.733855	.760240	.794130
Estimates for Experiment 2					
$\hat{\pi}_{RR}$	.600000	.600000	.609969	.639786	.638763
$\hat{\pi}_{RW}$	.466667	.466667	.470414	.450224	.455121
$\hat{\pi}_{WR}$	.233333	.233333	.183642	.131376	.099751
$\hat{\pi}_{WW}$	.666667	.666667	.692585	.740086	.757680
Estimates for Experiment 3					
$\hat{\pi}_{RR}$	.666667	.666667	.698107	.734338	.744597
$\hat{\pi}_{RW}$	.533333	.533333	.540123	.549716	.564636
$\hat{\pi}_{WR}$	.166667	.166667	.134130	.099008	.078710
$\hat{\pi}_{WW}$	.633333	.633333	.658355	.688965	.717075

$\hat{\pi}_{RR} = logit^{-1}(\hat{\beta}_0)$; $\hat{\pi}_{RW} = logit^{-1}(\hat{\beta}_0 + \hat{\beta}_2)$; $\hat{\pi}_{WR} = logit^{-1}(\hat{\beta}_0 + \hat{\beta}_1)$;
$\hat{\pi}_{WW} = logit^{-1}(\hat{\beta}_0 + \hat{\beta}_1 + \hat{\beta}_2 + \hat{\beta}_3)$

probability estimates from Laplace and Gibbs Sampler were practically indifferent, however Laplace confidence intervals will be narrower than those of Gibbs as seen with those of the β parameters.

4.2 Simulation Study Results

From Chapter 3 and the analyses given in Section 4.1, it is apparent that use of the Laplace approximation does not give up precision in computing MLEs and Likelihood Ratio Statistics in exchange for speedier computing times. The error in the Laplace approximate likelihood appears to be relatively flat, yielding desired maximum likelihood results with less computing time than numerical integration. It was also seen that REML yielded very similar estimates and inferences as Laplace, with larger differences observed in the Salamander Mating experiments, but with very minimal computing effort. The Cell Irradiation and Revertant Colony data sets are rather ideal for REML in that binomial indices and counts are large for the Cell Irradiation data set and the counts are greater than 20 for the Revertant Colony data set. For the Salamander Mating experiments, the data are Bernoulli counts, which should be less than ideal for REML estimation and inference.

To better understand how the estimates and inferences behave, a small simulation study was carried out. The focus of the study was on the potential bias in the estimators, their sampling distribution, the adequacy of the test statistics' assumed distributions, and correlations amongst the methods' estimates and test statistics. Using the Cell Irradiation experimental setup, four scenarios were simulated to investigate the behaviors of the estimates and test statistics.

4.2.1 The Simulation Design and Sample Generation

Recall that the Cell Irradiation experimental setup had 9 trials with 3 dishes placed in a chamber on each trial; i.e., 27 observations grouped into 9 sets of 3. The trial dispersion component, σ_1^2, was fixed at .25 while the dish dispersion component, σ_2^2, was fixed at .1 (the estimates from the Cell Irradiation

analysis were .2 and .01).[13] Recall as well that the observations were counts of surviving cells out of the 400 placed on each dish and that the data suggested a survival rate near 32%. These two items (number of cells — m, and true survival rate — π) were the only characteristics changed among the four simulations in an effort to understand what happens when conditions are less attractive to REML and IGLS. The particular parameter settings used for the simulations were:

Scenario	σ_1^2	σ_2^2	m	π	$\beta_0 (= logit(\pi))$
1	.25	.1	400	.32	-0.75
2	.25	.1	400	.1	-2.2
3	.25	.1	10	.32	-0.75
4	.25	.1	10	.1	-2.2

Of the four scenarios, it was felt that (1) and (2) would still be favorable for REML; that (3) may not be too bad for REML since $\pi \in (.2, .8)$; and that (4) may be unfavorable for REML since both m and π are small.

For each of the four scenarios, 200 sample data sets were generated under the design structure with m, π, σ_1^2 and σ_2^2 fixed appropriately. (For scenarios 2, 3, and 4, a second set of 200 data sets were generated after noting that their first runs each used identical starting seeds. For scenario 4, a third run of 200 data sets was generated after observing peculiarities with regards to Type I errors.) In each data set, the 9 trial (u_1) and 27 dish (u_2) random normal deviates were generated from separate generation streams. Each binomial observation y_{ij} (on dish j in trial i) was computed as the sum of m Bernoulli values generated with probability of success being $logit^{-1}(\beta_0 + u_{1i} + u_{2i})$, where u_{1i} is the i^{th} generated trial effect and u_{2ij} is the j^{th} dish effect in the i^{th} trial. That is, using the same matrices A_1 and A_2 from the Cell Irradiation model, the vector of binomial counts were created as the marginal row sum of a generated $27 \times m$ matrix of Bernoulli values with common row probabilities set to $logit^{-1}(\beta_0 + A_1 u_1 + A_2 u_2)$.

[13]This choice of the dispersion component settings was made to give similar experimental variation to that in the data, but also to increase the magnitude of the dish variation. Simulated data sets 1 and 2 in Section 3.3.2.2 used similar parameter settings to these here. When the dish variation was .01, it was noted that virtually no error occurs in using Laplace for ML estimation and inferences. It is of interest, however, to compare methods where the dish variation is non-trivial.

4.2.2 Presentation of the Simulation Study

The results from the simulations are presented in both graphical and tabular forms. Figures 6-27 (see pages 83-114) present results from the four scenarios, with plots of β estimates and test statistics (z-scores). These figures assist in visualizing correlations among methods, biases in estimates and inferences, and sampling distributions of estimates and z-scores. Tables 20-30 (see pages 85-115) give additional summary measures, quantifying various aspects of the simulation scenarios.

The figures for β estimates are composed of a matrix of scatter plots and a row of histograms of the estimates from the four methods: QL, IGLS, REML, and Maximum Likelihood via Laplace. Vertical and horizontal reference lines are drawn at the true β value, β_0. A row of histograms for each method's β estimate, overlaid with a normal curve centered at β_0 and variance being the sample variance, is given below the scatterplots. Figures of the tests statistics are identical in form. For QL, IGLS and REML, plotted are the respective z-scores for testing H_0: $\beta = \beta_0$ *vs.* H_a: $\beta \neq \beta_0$ (see Section 2.4). For the Laplace test scores, log-likelihood ratio statistics for testing H_0: $\beta = \beta_0$ *vs.* H_a: $\beta \neq \beta_0$ were converted to z-scores (i.e., $sign[\hat{\beta} - \beta_0] * \sqrt{\lambda}$; where λ is defined in Section 3.2.3). Reference lines in these plots are drawn at 0. Histograms of the z-scores, overlaid with the $N(0, 1)$ curve, are given below the scatterplots.

Following each pair of figures are tables of summary measures from the respective simulation scenario. For both β estimates and z-scores, standard sample summary measures are given for each method:

> Minimum (min), Maximum (max), Median (med), Mean (avg), Standard Deviation (sd), Bias (bias = avg $- \beta_0$), , and Mean Squared Error (mse = $\text{sd}^2 + \text{bias}^2$), and 95% Confidence Interval endpoints (avg $\pm$ 1.96sd/$\sqrt{\text{number of samples}}$).

Additionally, results from using a 5% significance test are given as:

> number of rejections (both $z \leq$ -1.96 and $z \geq$ 1.96), nominal Type I error rate ($\frac{\#\text{ rejections}}{\#\text{ samples}}$), and 95% Confidence Interval on the Type I error rate.[14]

[14]The confidence intervals for the error rate are found assuming the number of rejections are the result of independent Bernoulli trials with common probability of a rejection (the true Type I error rate) and are exact binomial confidence interval endpoints.

4.2.3 Summary of the Simulation Study

For the first two scenarios, there were very little differences seen between Laplace and REML results. The β estimates were virtually identical and apparently unbiased, test statistics (z-scores) were very similar–although falling on a sigmoidal curve with no variation about the curve. The sampling distribution of β and z values appeared approximately normal, Type I error rates were near 5%, and dispersion components estimates were very similar. Both Laplace and REML estimates and inferences appeared to be very good.

In scenario 3, subtle differences were observed between Laplace and REML results. The Laplace estimates were a bit more extreme, there was skewness seen in both methods' estimates and z-scores, and a less pronounced sigmoidal relationship between their z-scores with pronounced variation about the curve. The estimators' sampling distributions were slightly skewed with heavy tails to the extreme side, z-scores were slightly skewed to the negative side, more rejections coming from the negative side but Type I error rates still about 5%. Some discrepancies were seen with the estimation of dispersion components, especially with σ_2^2. The Laplace and REML estimates and inferences were still very similar, but not as similar as in scenarios 1 & 2.

In scenario 4, more pronounced differences were seen between Laplace and REML results. And it was the Laplace results that seemed to be the poorer values. Here there was substantial bias to the extreme in both β and z-scores for Laplace, substantial differences in estimates (both β and dispersion components) and z-scores, and slightly more skewness in Laplace than in REML values. There were more Laplace dispersion components estimated to 0 than for REML, yet Type I error rates maintained to be around 5%. For β estimates and z-scores, much more loss in correlation was seen between the methods, and the sigmoidal curve was all but lost. The z-scores for both were not severely skewed, but were shifted slightly negative, resulting in a lack of balance in rejections.

The results from REML and Laplace were very similar in all four scenarios, with discrepancies arising where they were expected—in scenario 4 mainly. And the lackluster performance in Laplace can be attributed to the bias in the β estimates. In conjunction with the bias, it was noticed that Laplace estimates became much different than REML estimates in scenarios 3 and 4. It

is possible that the optimization routine did not adequately converge to the MLEs in these scenarios, and that the dispersion components' MLEs are difficult to estimate here. The sampling distribution of the test statistics appeared to be adequately approximated by their assumed distributions.

The β estimates from QL and IGLS were identical in all four scenarios and were always closer to 0 than those of REML and Laplace (as seen in the analyses of the example data sets). These values tended to be biased towards 0 as well, and confidence interval endpoints typically were far from the true β value. The QL z-scores were typically larger than those of the other methods, implying standard errors were too small and giving high Type I errors rates. This was the case as well for IGLS, although to a much lesser extent. The QL method did not perform well at all in the simulations, and IGLS did not have favorable results either—being often biased and rejecting too often.

It should be noted that 2-node Gaussian quadrature maximum likelihood analyses were done on the simulated data sets as well. In all scenarios, these results were identical to those of Laplace. In scenario 4, discrepancies in the dispersion component estimation was seen, leading to the thought that either the MLEs were not found, or that there is too little information to find them. The numerical integration results were not plotted in the figures as they added little information.

Figure 6. Cell Irradiation Scenario 1: β-estimates

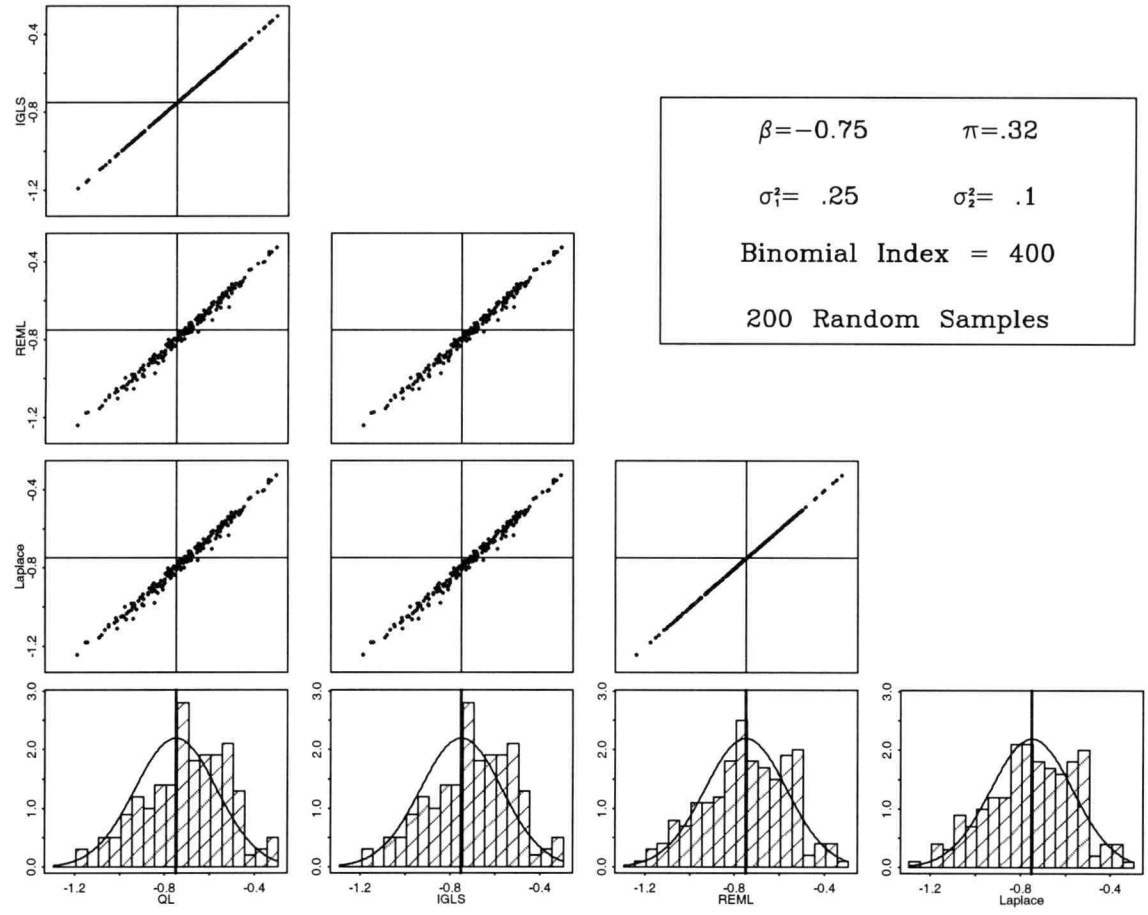


Figure 7. Cell Irradiation Scenario 1: z-statistics for H_0 $\beta = -0.75$

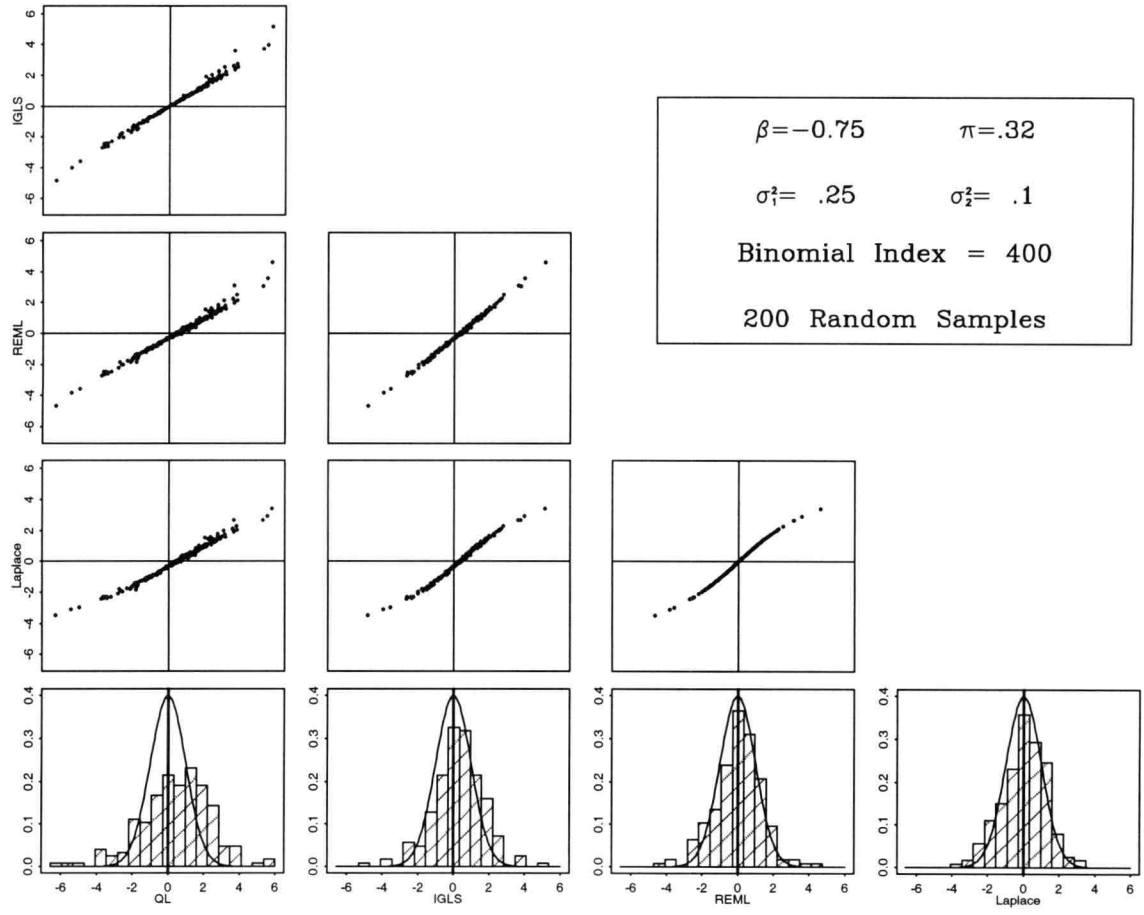


Table 20. Summary of Simulation Results for Cell Irradiation Data Set Scenario 1

True β = -.75, Binomial Index = 400, 200 samples

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-1.190	-.306	-.694	-.699	.182	.051	.0356	-.724	-.674
IGLS	-1.190	-.306	-.694	-.699	.182	.051	.0356	-.724	-.674
REML	-1.240	-.326	-.743	-.746	.190	.004	.0362	-.772	-.719
Laplace	-1.244	-.327	-.745	-.748	.191	.002	.0364	-.774	-.722

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-6.325	5.772	.568	.472	1.939	.472	3.7593	.203	.741
IGLS	-4.841	5.130	.348	.326	1.399	.326	1.9584	.132	.520
REML	-4.679	4.613	.047	.022	1.311	.022	1.7190	-.159	.204
Laplace	-3.462	3.410	.037	.018	1.213	.018	1.4714	-.150	.186

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	19	46	32.5 %	26.5 %	39.5 %
IGLS	12	18	15.0 %	10.8 %	20.7 %
REML	13	10	11.5 %	7.8 %	16.8 %
Laplace	11	9	10.0 %	6.6 %	15.0 %

Figure 8. Cell Irradiation Scenario 2, Set 1: β-estimates

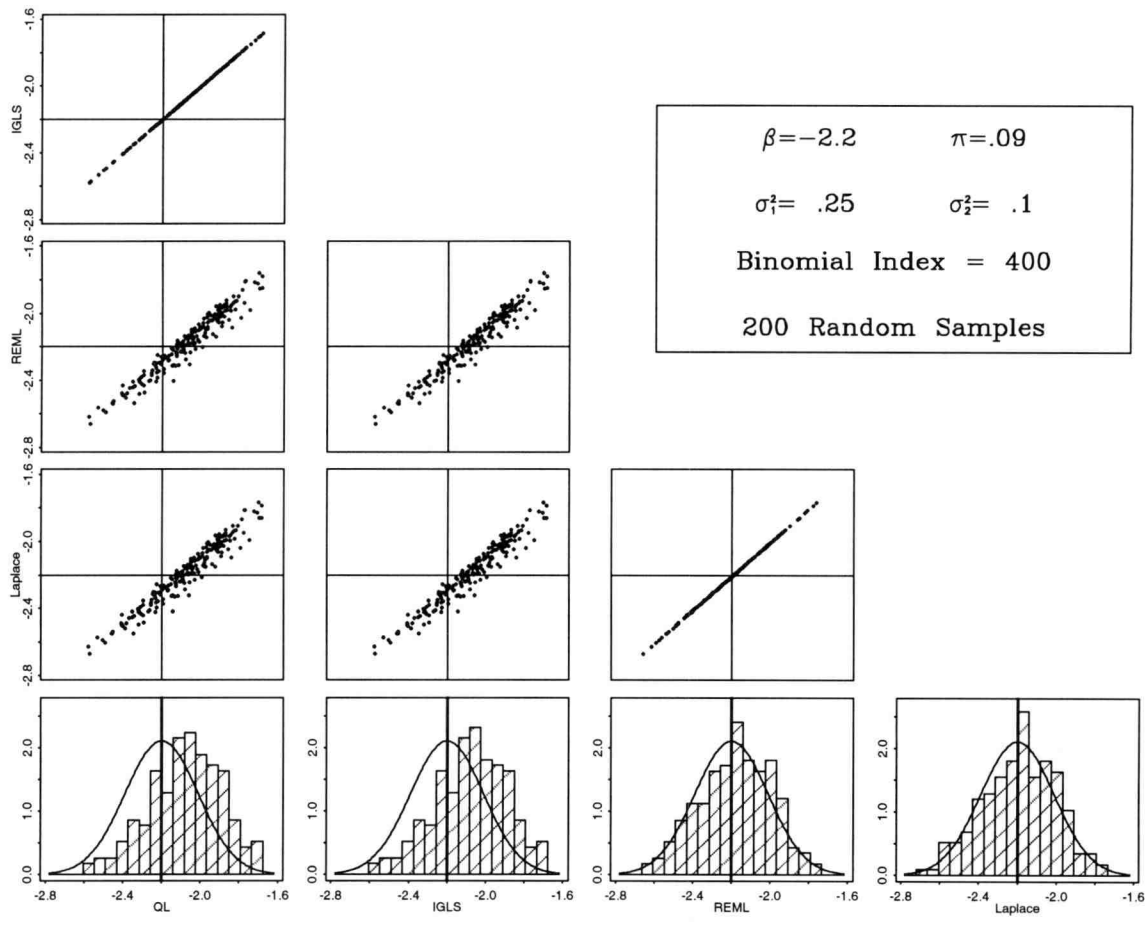


Figure 9. Cell Irradiation Scenario 2, Set 1: z-statistics for H_0 $\beta = -2.2$

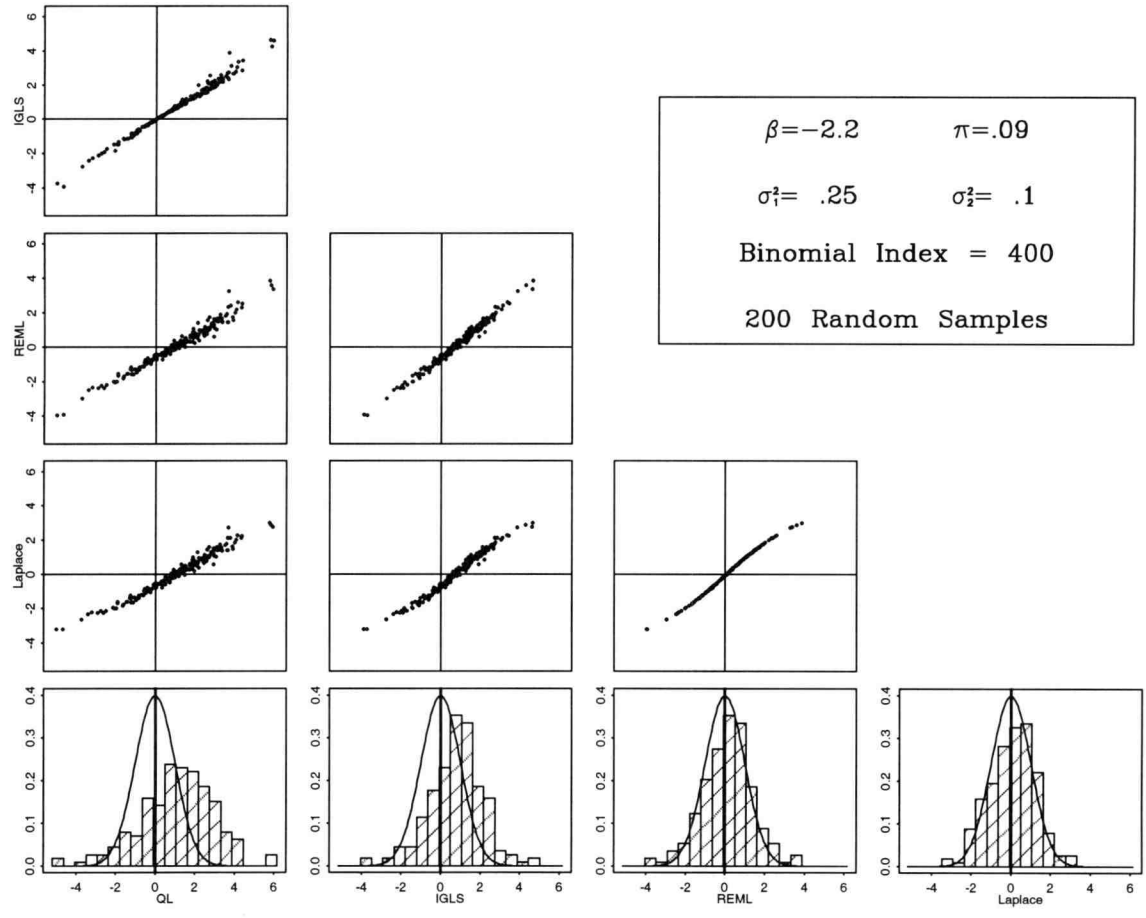


Table 21. Summary of Simulation Results for Cell Irradiation Data Set Scenario 2, Set 1

True β = -2.2, Binomial Index = 400, 200 samples

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.580	-1.685	-2.065	-2.076	.189	.124	.0511	-2.102	-2.050
IGLS	-2.580	-1.685	-2.065	-2.076	.189	.124	.0511	-2.102	-2.050
REML	-2.659	-1.763	-2.175	-2.180	.186	.020	.0349	-2.206	-2.154
Laplace	-2.672	-1.769	-2.184	-2.189	.187	.011	.0352	-2.215	-2.163

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-4.981	5.924	1.165	1.109	1.871	1.109	4.7295	.850	1.368
IGLS	-3.899	4.643	.801	.779	1.381	.779	2.5125	.587	.970
REML	-3.972	3.848	.143	.130	1.270	.130	1.6298	-.046	.306
Laplace	-3.201	3.008	.091	.071	1.191	.071	1.4228	-.094	.234

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	12	67	39.5 %	33.2 %	46.6 %
IGLS	7	37	22.0 %	16.9 %	28.4 %
REML	12	11	11.5 %	7.8 %	16.8 %
Laplace	12	10	11.0 %	7.4 %	16.2 %

Figure 10. Cell Irradiation Scenario 2, Set 2: β-estimates

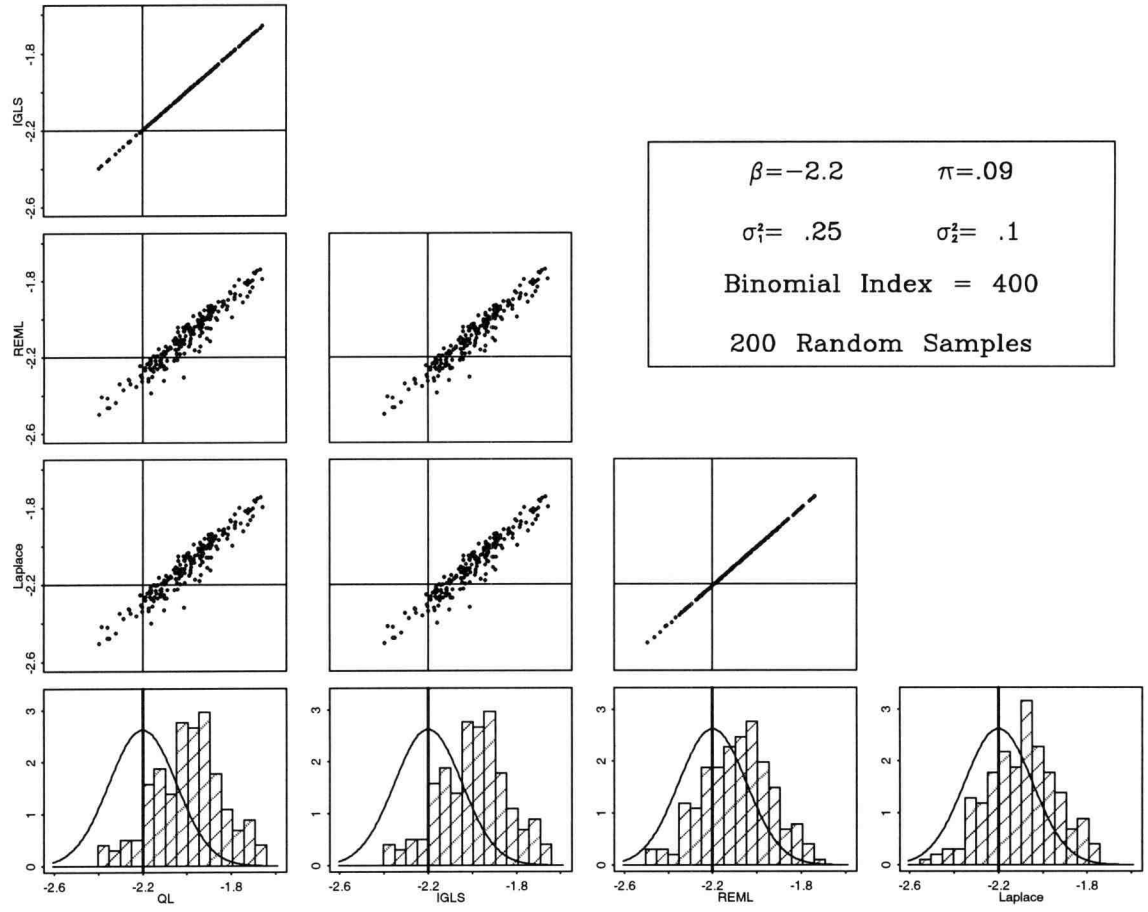


Figure 11. Cell Irradiation Scenario 2, Set 2: z-statistics for H_0 $\beta = -2.2$

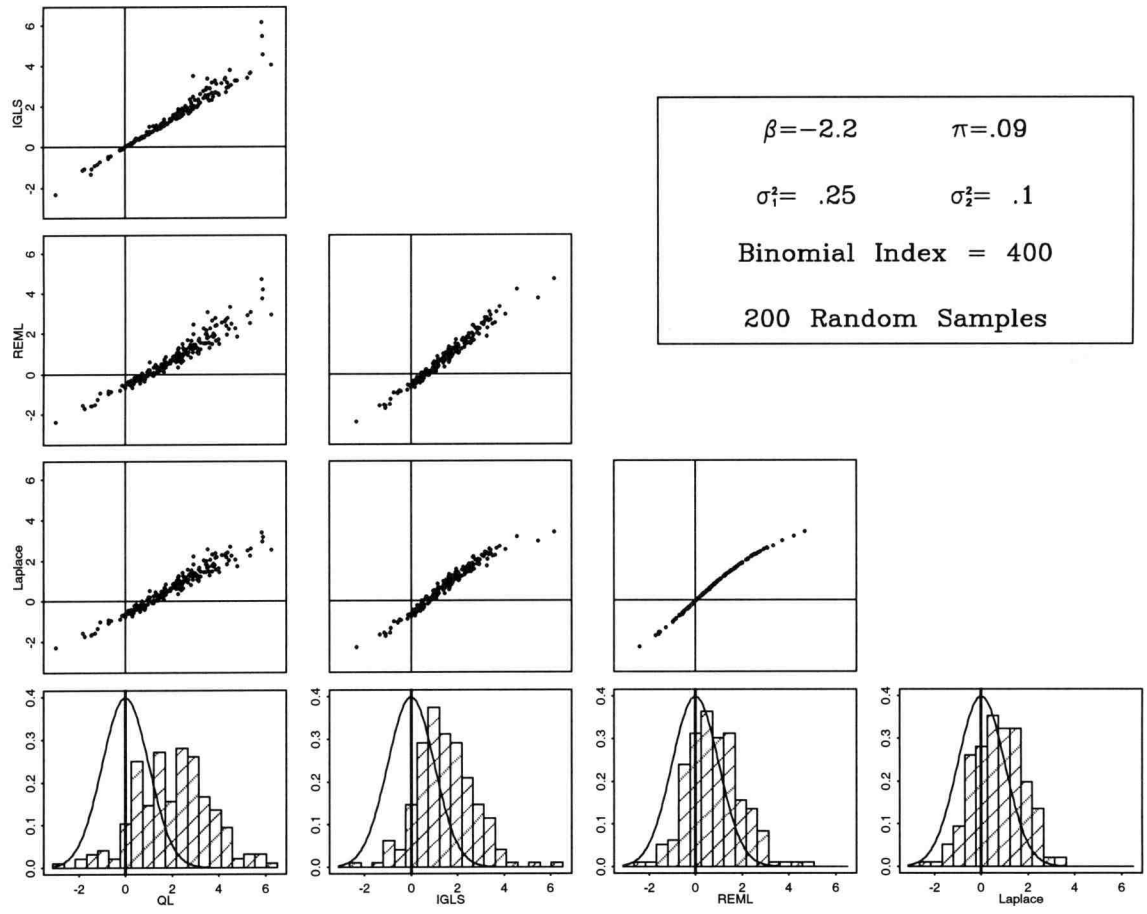


Table 22. Summary of Simulation Results for Cell Irradiation Data Set Scenario 2, Set 2

True β = -2.2, Binomial Index = 400, 200 samples (2^{nd} Set)

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.398	-1.657	-1.984	-1.994	.152	.206	.0657	-2.005	-1.963
IGLS	-2.398	-1.656	-1.985	-1.994	.152	.206	.0657	-2.005	-1.963
REML	-2.496	-1.739	-2.071	-2.088	.156	.112	.0370	-2.110	-2.066
Laplace	-2.505	-1.745	-2.078	-2.096	.157	.104	.0356	-2.118	-2.074

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.996	6.264	2.169	2.047	1.601	2.047	6.7536	1.825	2.269
IGLS	-2.334	6.159	1.442	1.467	1.221	1.467	3.6439	1.298	1.636
REML	-2.397	4.697	.712	.795	1.140	.795	1.9307	.637	.953
Laplace	-2.288	3.393	.687	.696	1.046	.696	1.5786	.551	.841

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	1	107	54.0 %	47.3 %	61.1 %
IGLS	1	62	31.5 %	25.6 %	38.4 %
REML	1	29	15.0 %	10.8 %	20.7 %
Laplace	1	25	13.0 %	9.1 %	18.5 %

Figure 12. Cell Irradiation Scenario 2, Sets 1 & 2: β-estimates

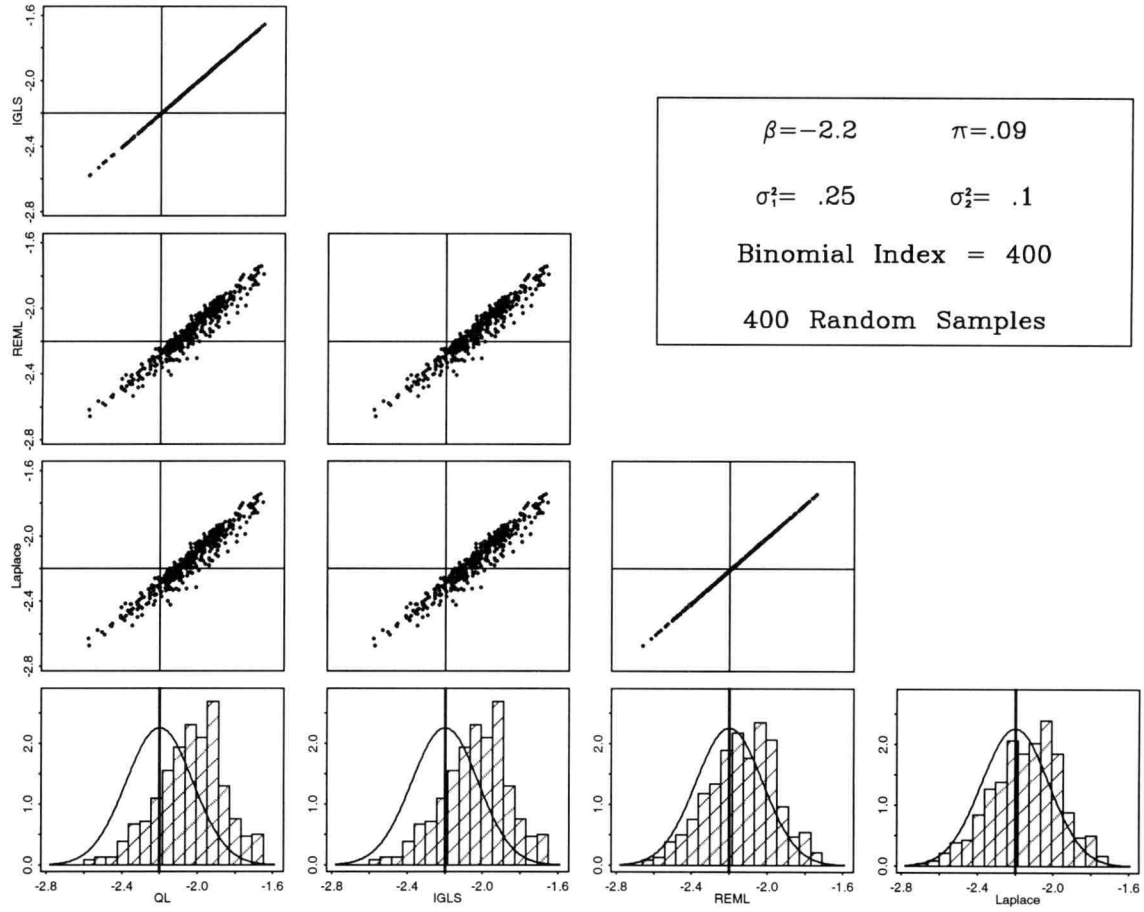


Figure 13. Cell Irradiation Scenario 2, Sets 1 & 2: z-statistics for H_0 $\beta = -2.2$

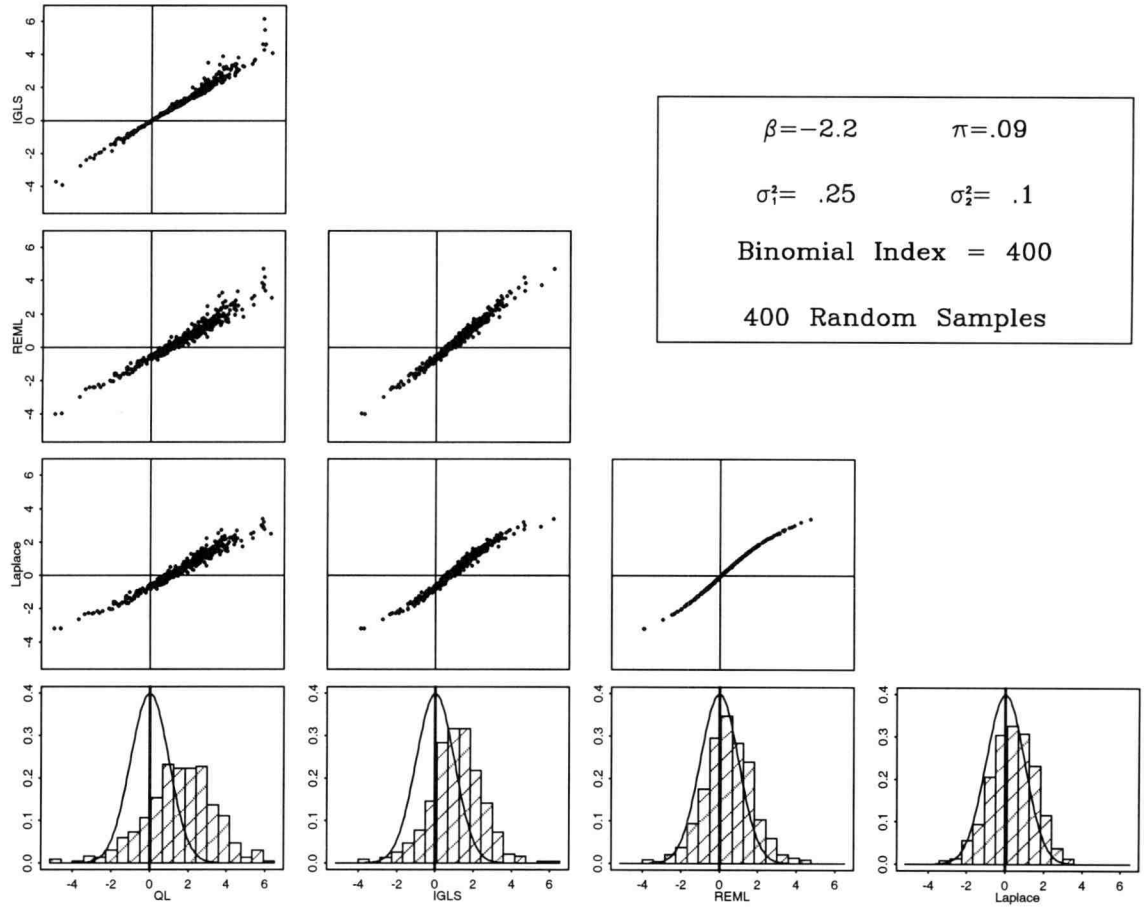


Table 23. Summary of Simulation Results for Cell Irradiation Data Set Scenario 2, Sets 1 & 2 Combined

True β = -2.2, Binomial Index = 400, 400 samples (Combined)

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.580	-1.657	-2.021	-2.035	.176	.165	.0582	-2.052	-2.018
IGLS	-2.580	-1.657	-2.021	-2.035	.176	.165	.0582	-2.052	-2.018
REML	-2.659	-1.739	-2.127	-2.134	.178	.066	.0360	-2.151	-2.117
Laplace	-2.672	-1.745	-2.137	-2.142	.179	.058	.0354	-2.160	-2.125

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-4.981	6.260	1.645	1.578	1.801	1.578	5.7337	1.402	1.368
IGLS	-3.899	6.159	1.150	1.123	1.347	1.123	3.0755	.991	1.255
REML	-3.972	4.697	.453	.462	1.250	.462	1.7759	.340	.585
Laplace	-3.201	3.393	.419	.383	1.162	.383	1.4969	.269	.497

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	13	174	46.8 %	42.0 %	51.8 %
IGLS	8	99	26.8 %	22.7 %	31.4 %
REML	13	40	13.3 %	10.3 %	17.0 %
Laplace	13	35	12.0 %	9.2 %	15.6 %

Figure 14. Cell Irradiation Scenario 3, Set 1: β-estimates

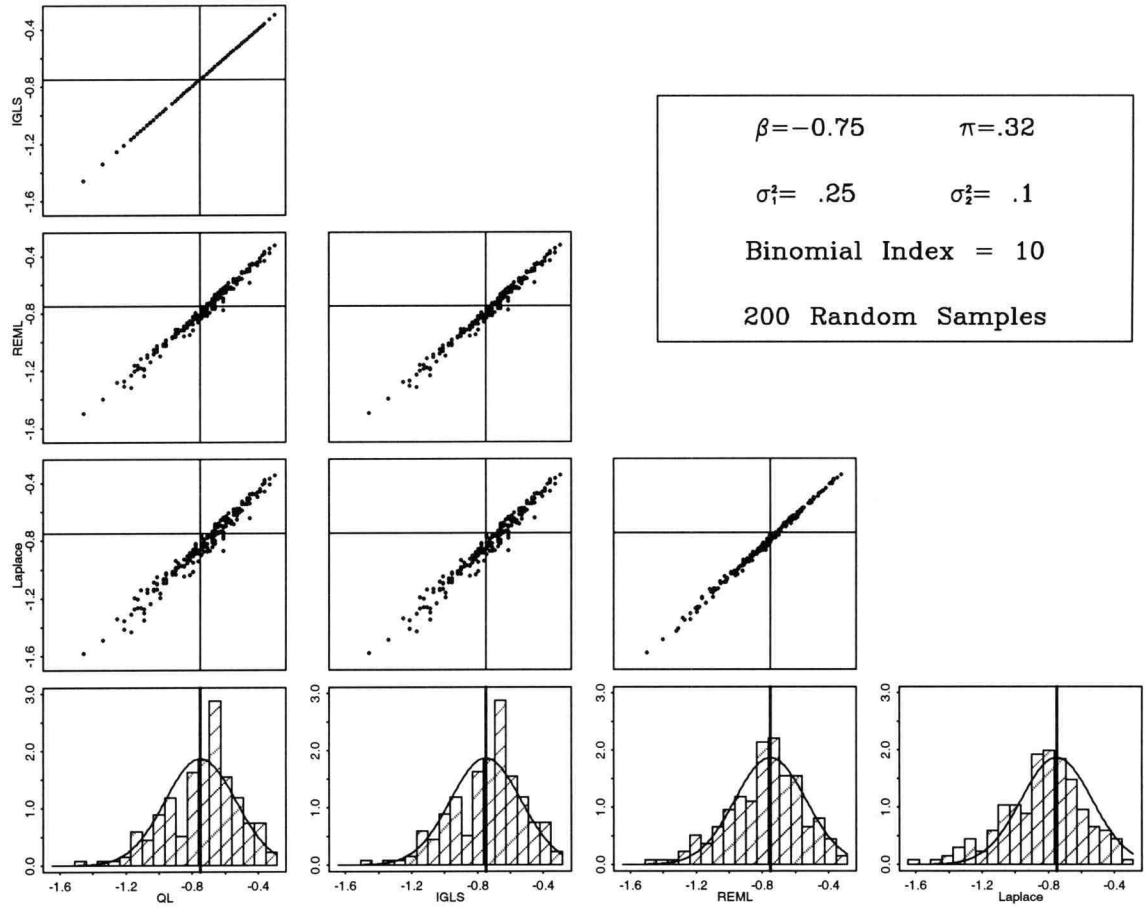


Figure 15. Cell Irradiation Scenario 3, Set 1: z-statistics for H_0 $\beta = -0.75$

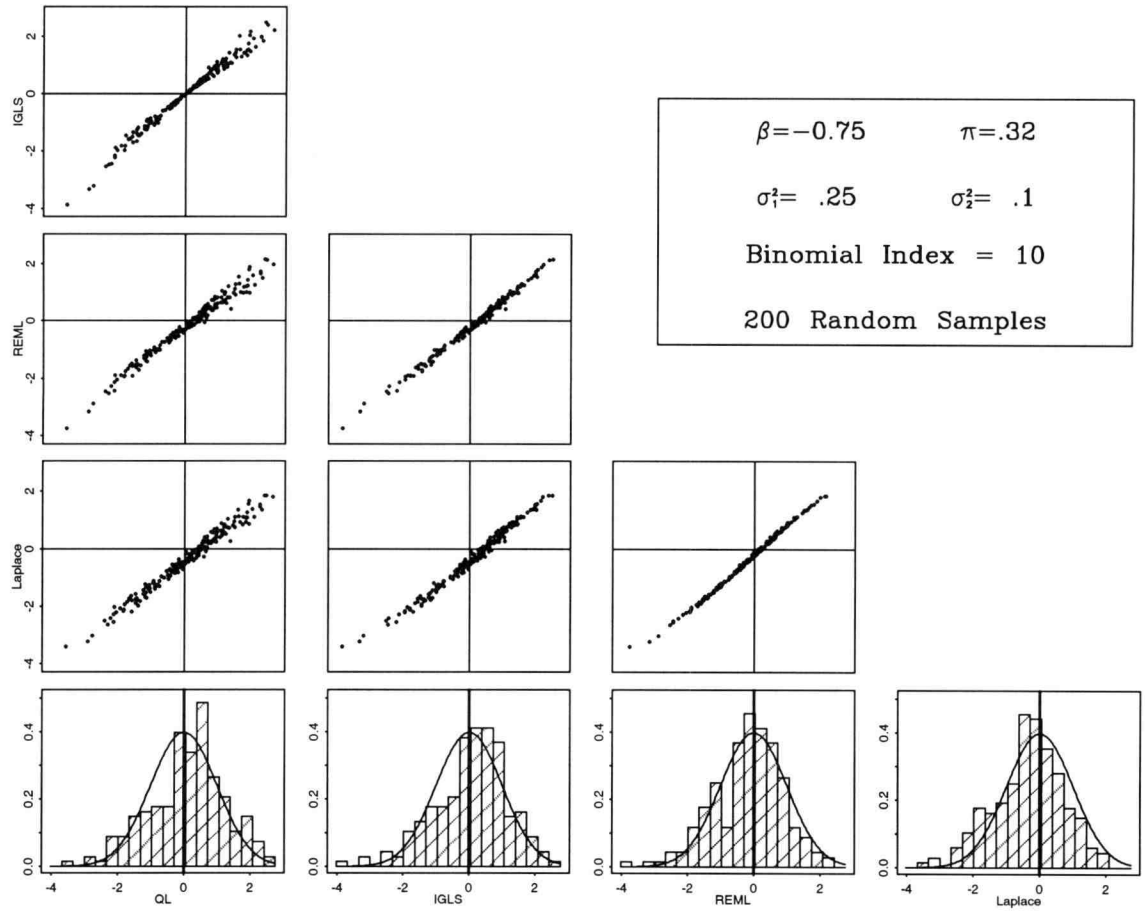


Table 24. Summary of Simulation Results for Cell Irradiation Data Set Scenario 3, Set 1

True β = -.75, Binomial Index = 10, 200 samples

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-1.457	-.299	-.710	-.735	.215	.015	.0463	-.765	-.705
IGLS	-1.457	-.299	-.710	-.735	.215	.015	.0463	-.765	-.705
REML	-1.500	-.323	-.763	-.784	.223	-.034	.0511	-.815	-.753
Laplace	-1.583	-.345	-.815	-.831	.240	-.081	.0639	-.864	-.798

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-3.570	2.661	.235	.109	1.159	.109	1.3542	-.051	.270
IGLS	-3.864	2.494	.237	.067	1.119	.067	1.2562	-.088	.222
REML	-3.759	2.160	-.041	-.153	1.038	-.153	1.1003	-.297	-.009
Laplace	-3.398	1.845	-.220	-.345	1.038	-.345	1.1960	-.488	-.201

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	10	8	9.0 %	5.8 %	13.9 %
IGLS	10	8	9.0 %	5.8 %	13.9 %
REML	9	3	6.0 %	3.5 %	10.3 %
Laplace	15	0	7.5 %	4.6 %	12.1 %

Figure 16. Cell Irradiation Scenario 3, Set 2: β-estimates

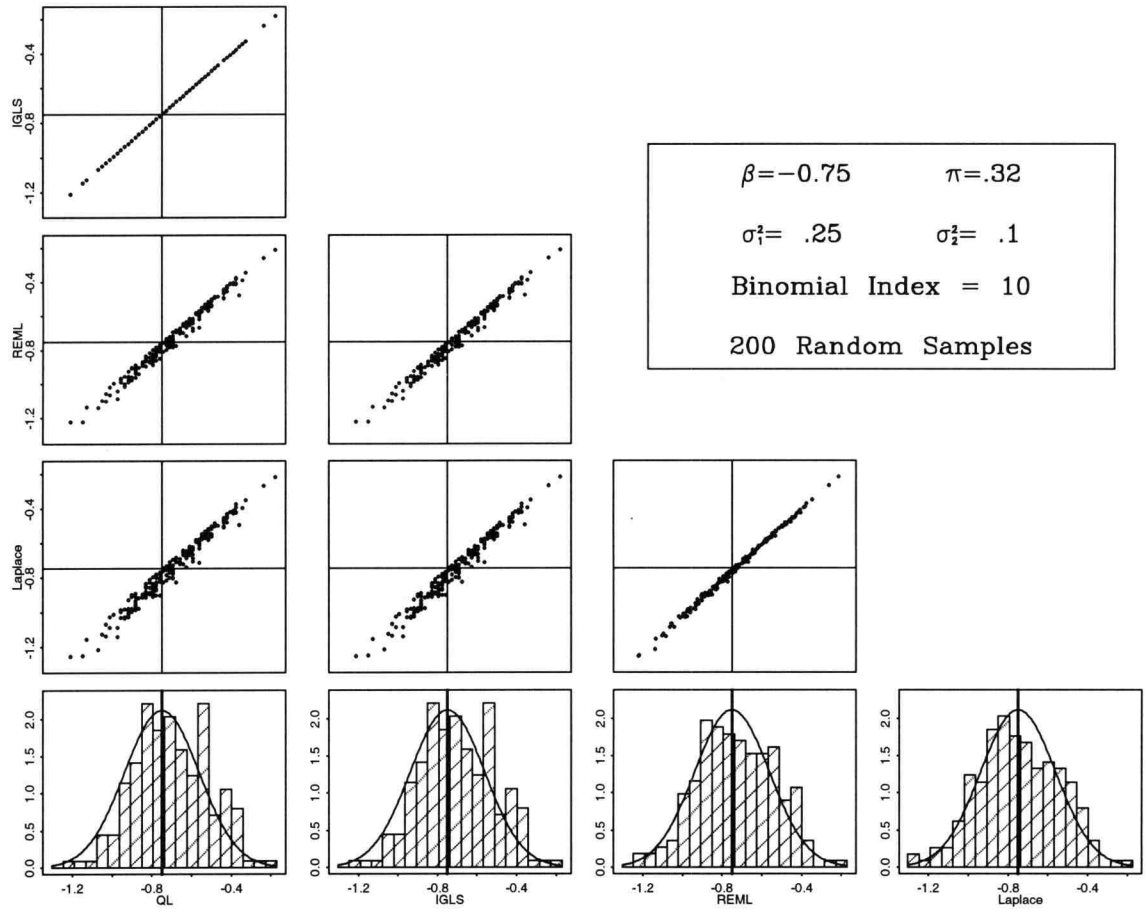


Figure 17. Cell Irradiation Scenario 3, Set 2: z-statistics for H_0 $\beta = -0.75$

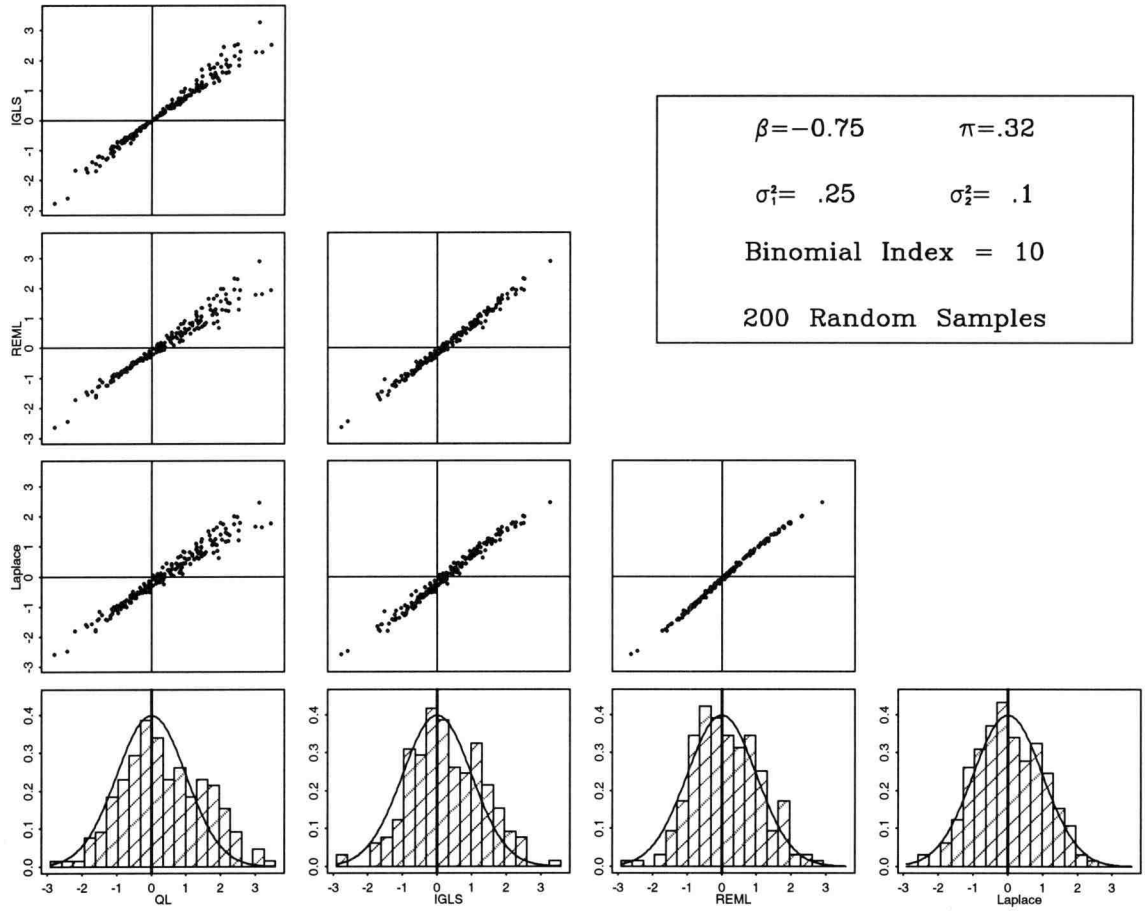


Table 25. Summary of Simulation Results for Cell Irradiation Data Set Scenario 3, Set 2

True β = -.75, Binomial Index = 10, 200 samples (2^{nd} Set)

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-1.210	-.178	-.710	-.698	.188	.052	.0382	-.724	-.672
IGLS	-1.210	-.178	-.710	-.698	.188	.052	.0382	-.724	-.672
REML	-1.222	-.214	-.737	-.728	.196	.022	.0388	-.756	-.701
Laplace	-1.254	-.219	-.755	-.747	.202	.003	.0407	-.775	-.719

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.794	3.438	.283	.359	1.199	.359	1.5650	.192	.525
IGLS	-2.773	3.258	.235	.299	1.051	.299	1.1936	.154	.445
REML	-2.637	2.894	.052	.140	.959	.140	.9400	.006	.274
Laplace	-2.574	2.461	-.024	.032	.952	.032	.9076	-.100	.164

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	3	20	11.5 %	7.8 %	16.8 %
IGLS	2	12	7.0 %	4.3 %	11.5 %
REML	2	5	3.6 %	1.8 %	7.2 %
Laplace	2	3	2.5 %	1.1 %	5.7 %

Figure 18. Cell Irradiation Scenario 3, Sets 1 & 2: β-estimates

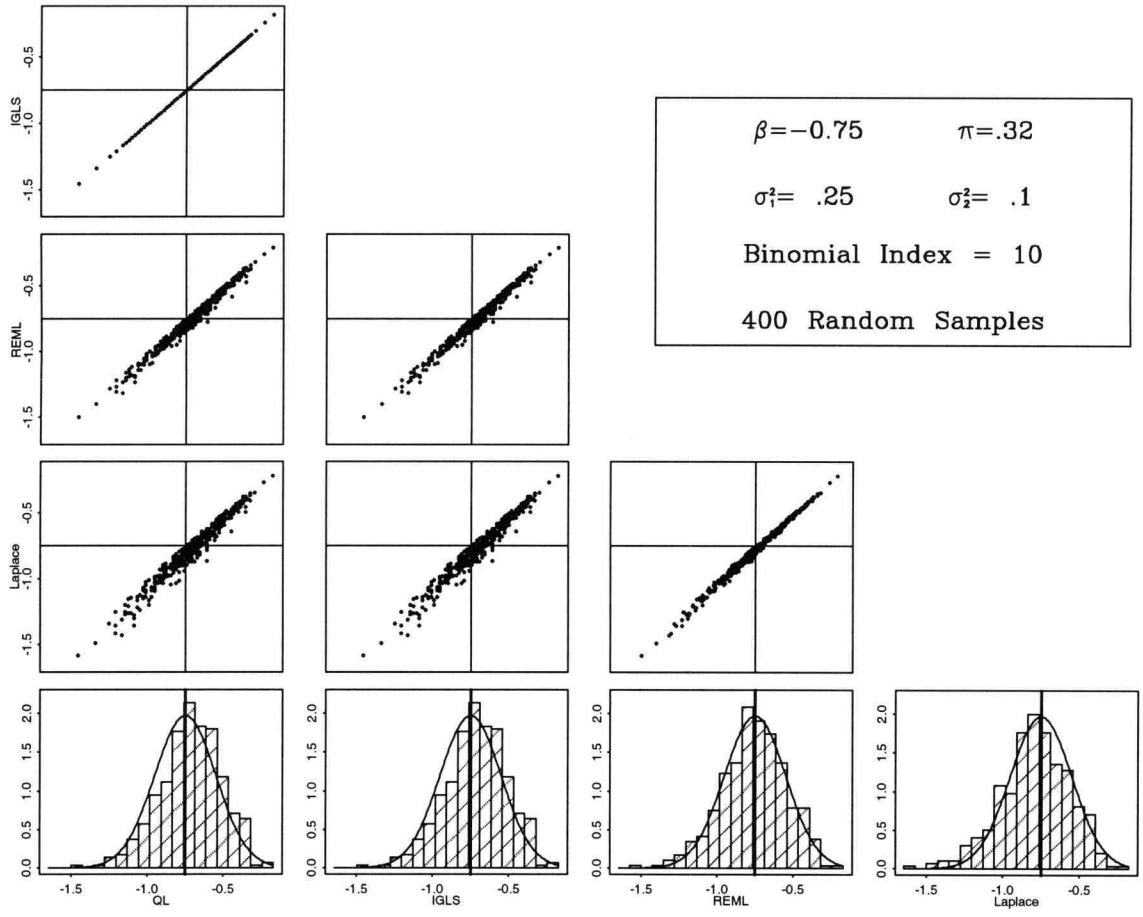


Figure 19. Cell Irradiation Scenario 3, Sets 1 & 2: z-statistics for H_0 $\beta = -0.75$

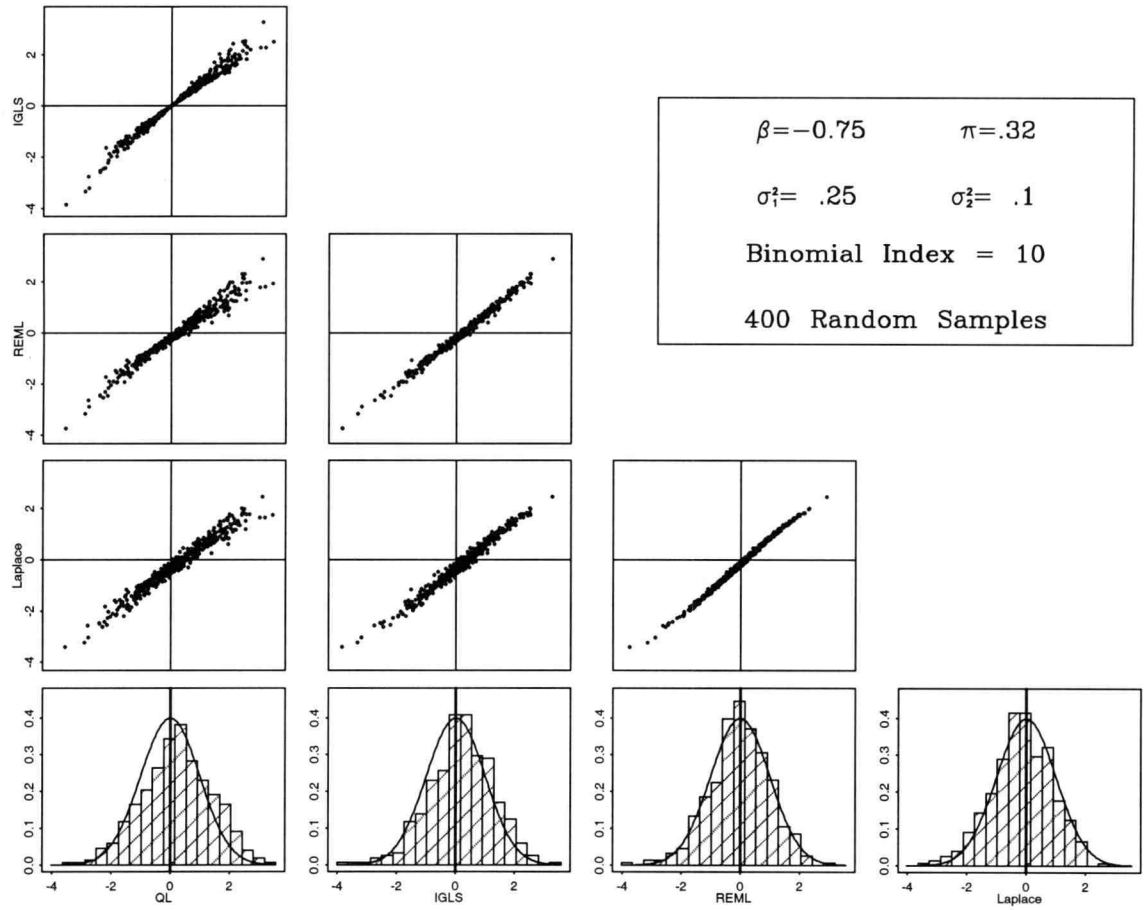


Table 26. Summary of Simulation Results for Cell Irradiation Data Set Scenario 3, Sets 1 & 2 Combined

True β = -.75, Binomial Index = 10, 400 samples (Combined)

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-1.457	-.178	-.710	-.716	.202	.034	.0420	-.736	-.696
IGLS	-1.457	-.178	-.710	-.716	.202	.034	.0420	-.736	-.696
REML	-1.500	-.214	-.752	-.757	.212	-.007	.0450	-.778	-.736
Laplace	-1.583	-.219	-.780	-.789	.225	-.039	.0522	-.811	-.767

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-3.570	3.438	.267	.234	1.184	.234	1.4566	.118	.350
IGLS	-3.864	3.258	.235	.183	1.090	.183	1.2216	.076	.290
REML	-3.759	2.894	-.010	-.008	1.009	-.008	1.0182	-.107	.091
Laplace	-3.398	2.465	-.144	-.156	1.012	-.156	1.0485	-.255	-.057

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	13	28	10.3 %	7.7 %	13.7 %
IGLS	12	20	8.0 %	5.8 %	11.1 %
REML	11	8	4.8 %	3.1 %	7.4 %
Laplace	17	3	5.0 %	3.3 %	7.6 %

Figure 20. Cell Irradiation Scenario 4, Set 1: β-estimates

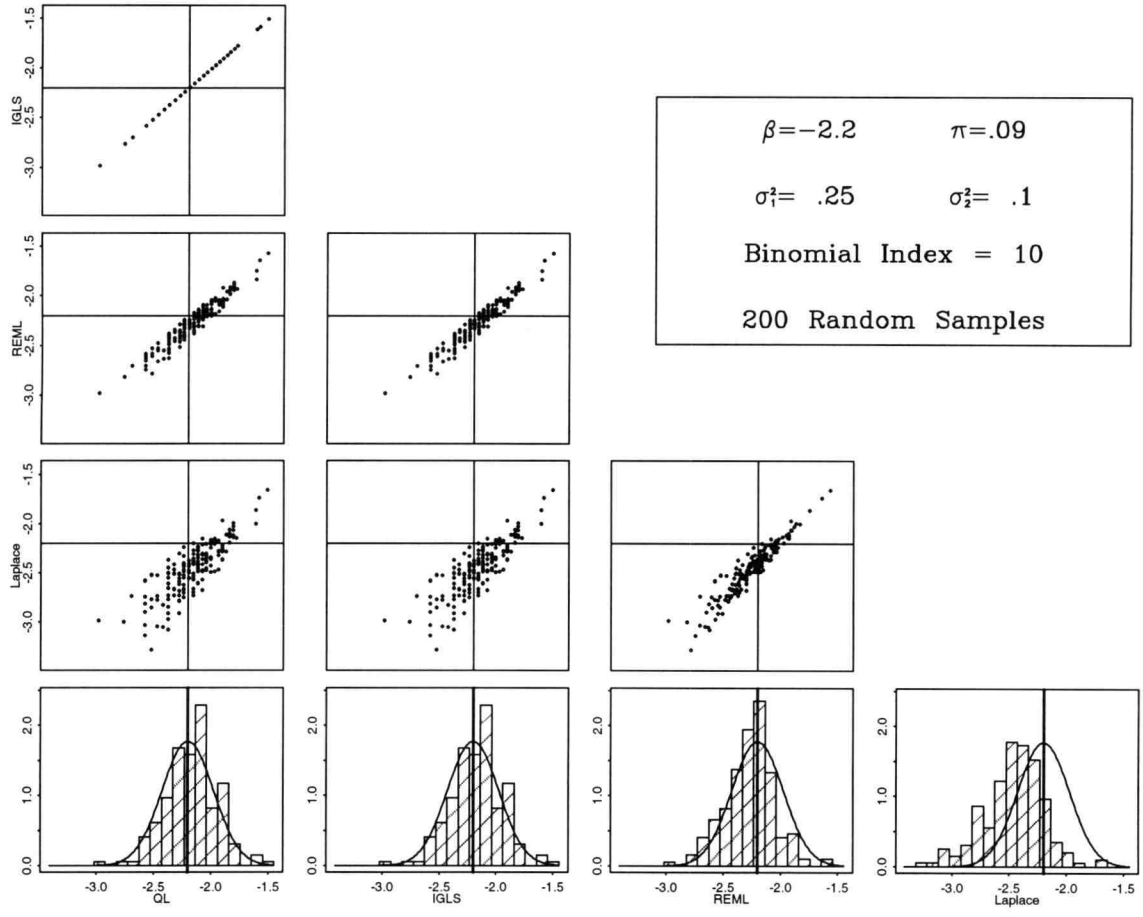


Figure 21. Cell Irradiation Scenario 4, Set 1: z-statistics for H_0 $\beta = -2.2$

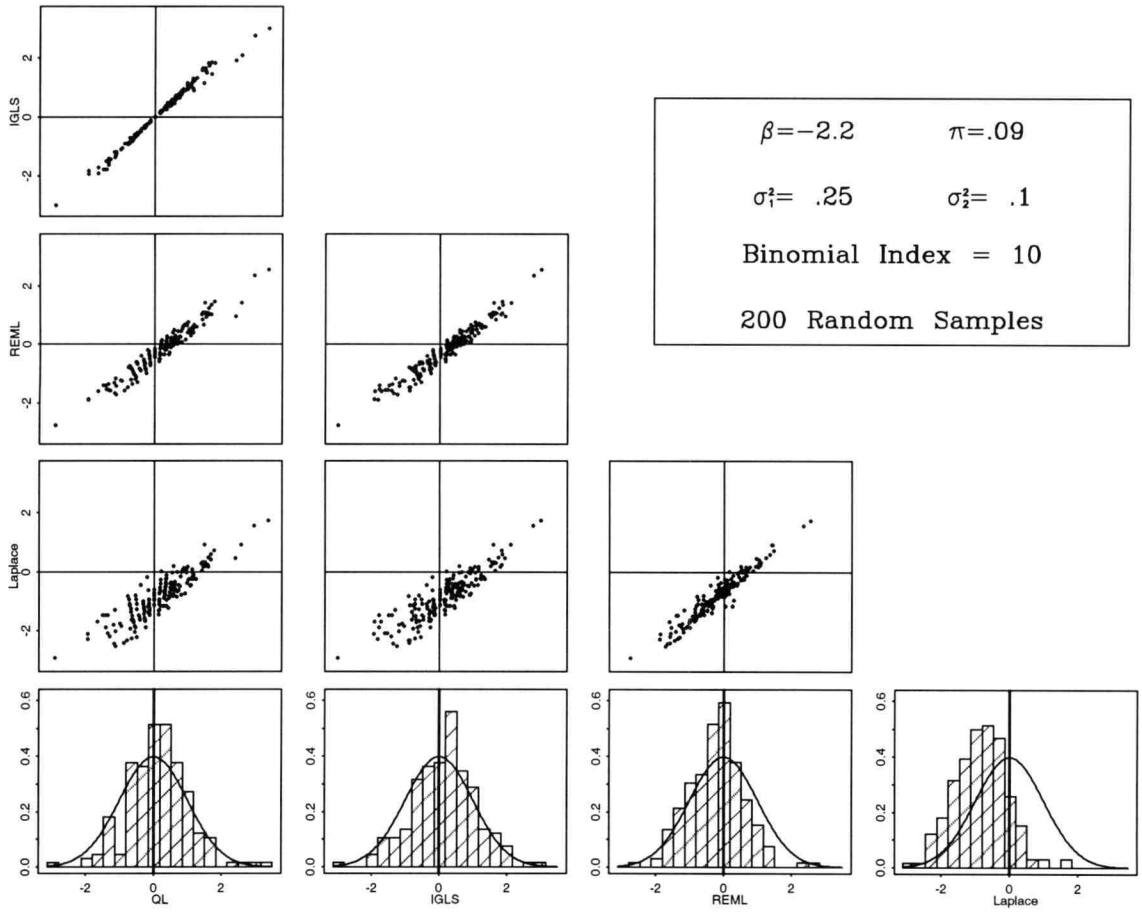


Table 27. Summary of Simulation Results for Cell Irradiation Data Set Scenario 4, Set 1

True β = -2.2, Binomial Index = 10, 200 samples

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.984	-1.506	-2.157	-2.166	.225	.034	.0519	-2.200	-2.130
IGLS	-2.984	-1.506	-2.157	-2.166	.225	.034	.0519	-2.200	-2.130
REML	-2.984	-1.571	-2.247	-2.265	.217	-.065	.0512	-2.300	-2.240
Laplace	-3.281	-1.663	-2.441	-2.462	.256	-.262	.1339	-2.498	-2.427

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.905	3.359	.178	.163	.890	.163	.8187	.039	.286
IGLS	-2.997	2.996	.200	.151	.926	.151	.8803	.023	.279
REML	-2.755	2.545	-.173	-.212	.791	-.212	.6700	-.322	-.103
Laplace	-2.923	1.739	-.813	-.836	.764	-.836	1.2837	-.942	-.731

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	1	4	2.5 %	1.1 %	5.7 %
IGLS	1	3	2.0 %	0.8 %	5.0 %
REML	1	2	1.5 %	0.6 %	4.3 %
Laplace	15	0	7.5 %	4.6 %	12.1 %

Figure 22. Cell Irradiation Scenario 4, Set 2: β-estimates

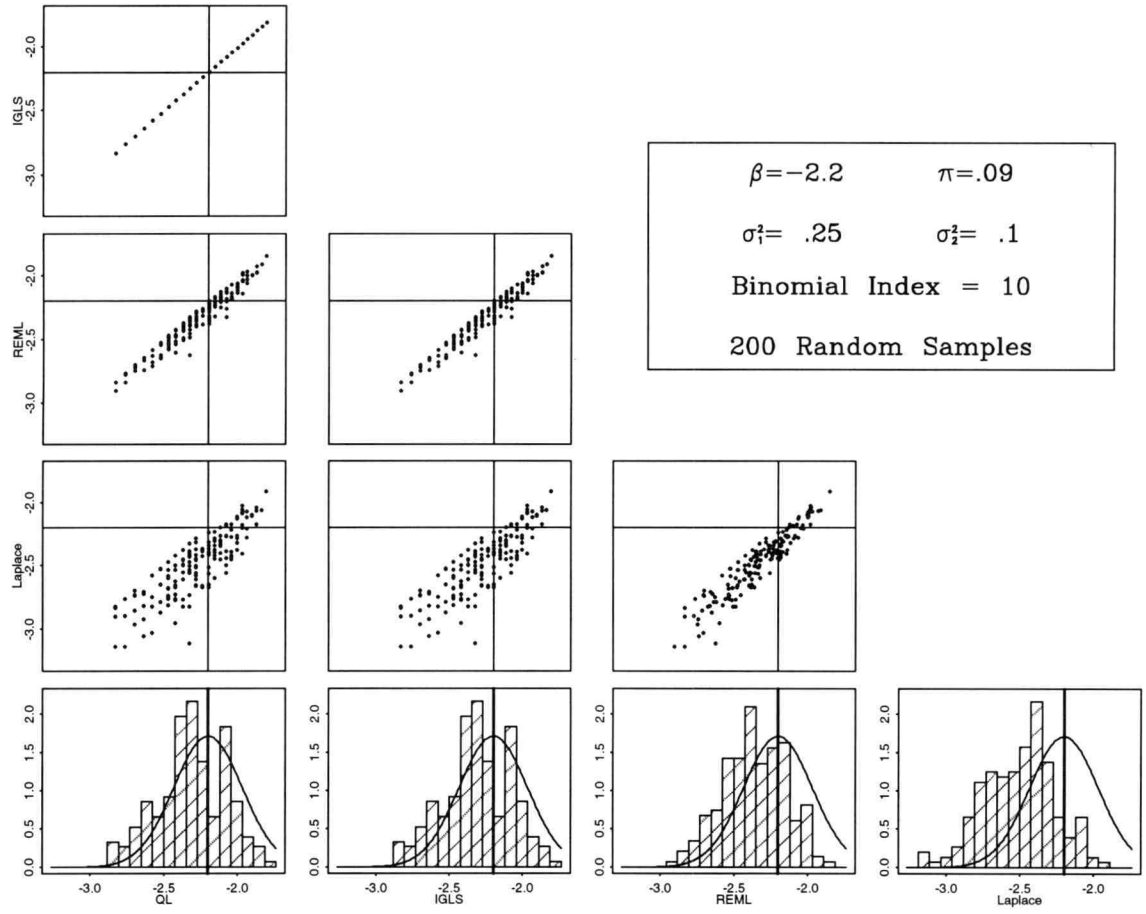


Figure 23. Cell Irradiation Scenario 4, Set 2: z-statistics for H_0 $\beta = -2.2$

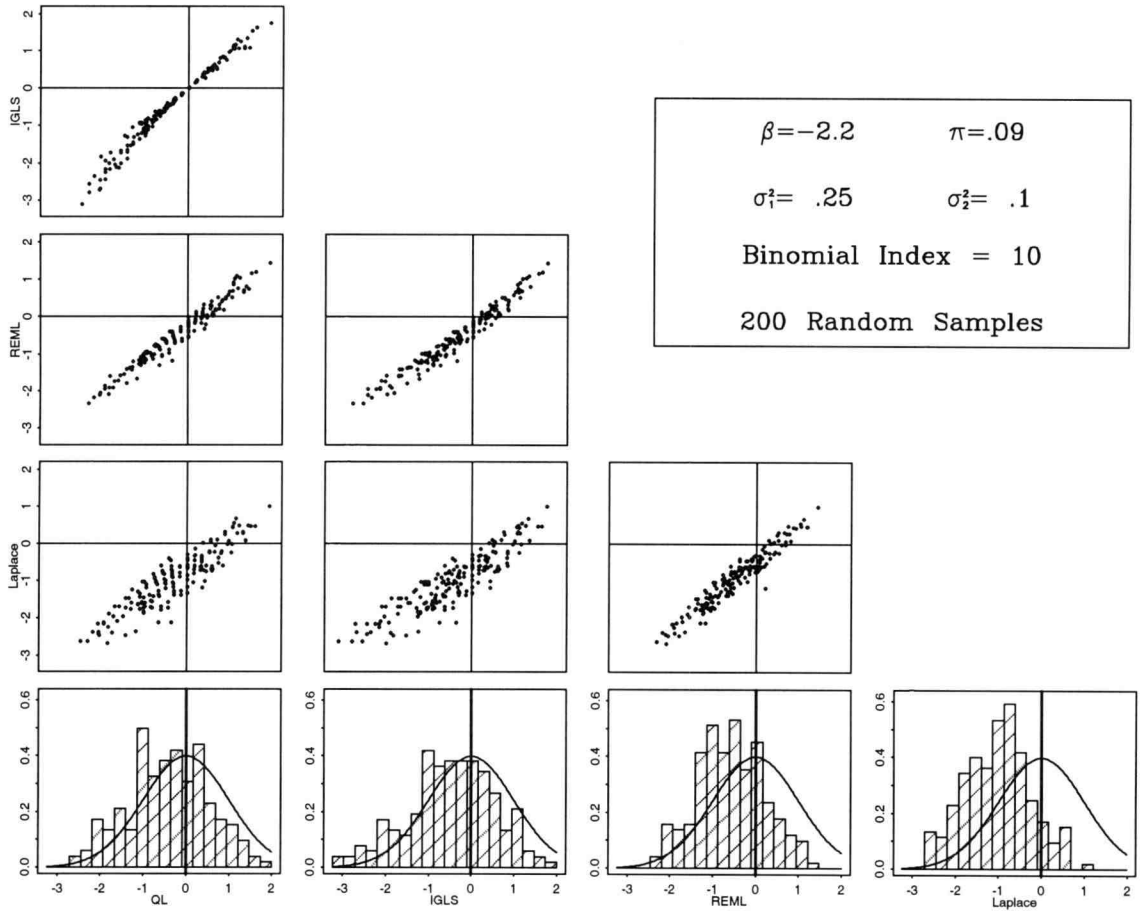


Table 28. Summary of Simulation Results for Cell Irradiation Data Set Scenario 4, Set 2

True β = -2.2, Binomial Index = 10, 200 samples (2^{nd} Set)

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.833	-1.809	-2.282	-2.309	.234	-.109	.0665	-2.341	-2.277
IGLS	-2.833	-1.809	-2.282	-2.309	.234	-.109	.0665	-2.341	-2.277
REML	-2.904	-1.850	-2.360	-2.359	.213	-.159	.0707	-2.389	-2.329
Laplace	-3.138	-1.914	-2.496	-2.502	.234	-.302	.1457	-2.534	-2.470

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.473	1.921	-.368	-.402	.923	-.402	1.0141	-.530	-.274
IGLS	-3.107	1.744	-.397	-.458	1.003	-.458	1.2168	-.597	-.319
REML	-2.323	1.434	-.595	-.568	.784	-.568	.9371	-.678	-.457
Laplace	-3.138	1.000	-1.031	-1.021	.765	-1.021	1.6277	-1.127	-.915

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	9	0	4.5 %	2.4 %	8.4 %
IGLS	19	0	9.5 %	6.2 %	14.4 %
REML	6	0	3.1 %	1.5 %	6.6 %
Laplace	23	0	11.5 %	7.8 %	16.8 %

Figure 24. Cell Irradiation Scenario 4, Set 3: β-estimates

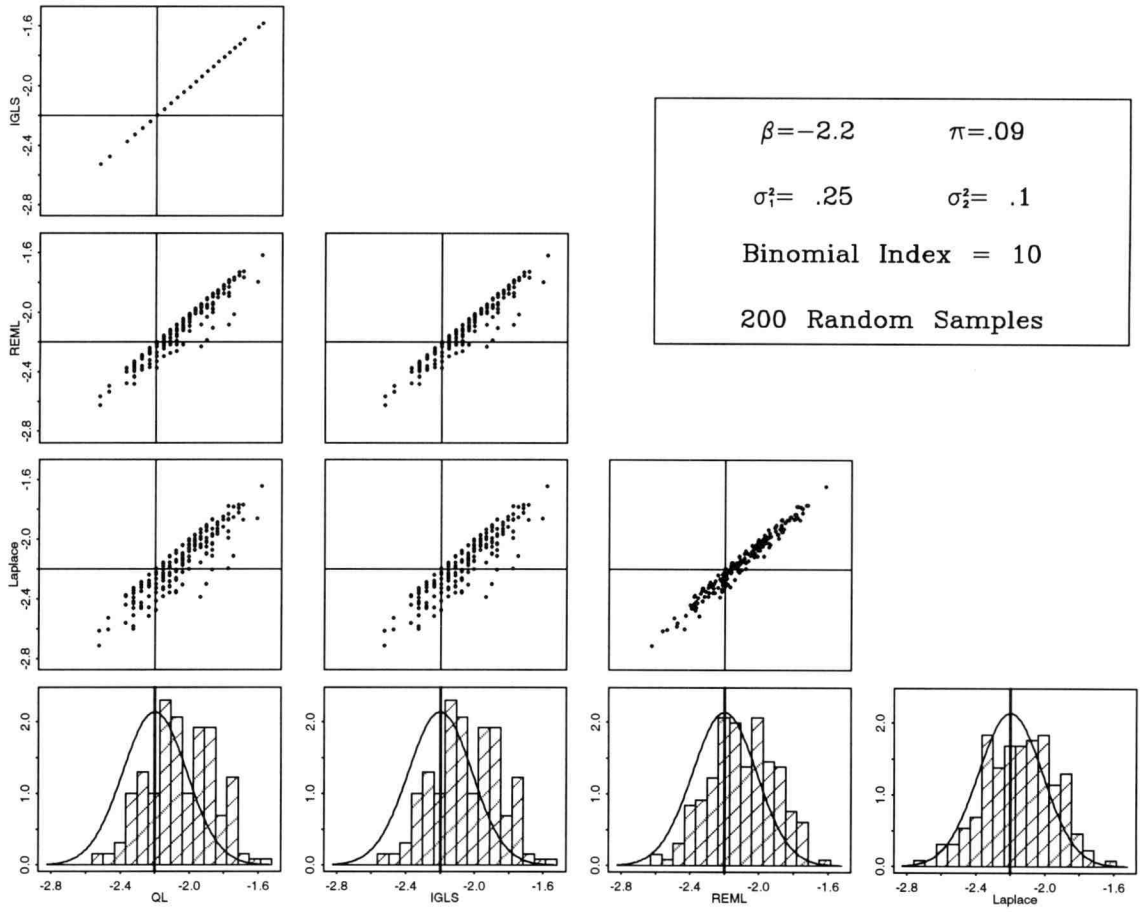


Figure 25. Cell Irradiation Scenario 4, Set 3: z-statistics for H_0 $\beta = -2.2$

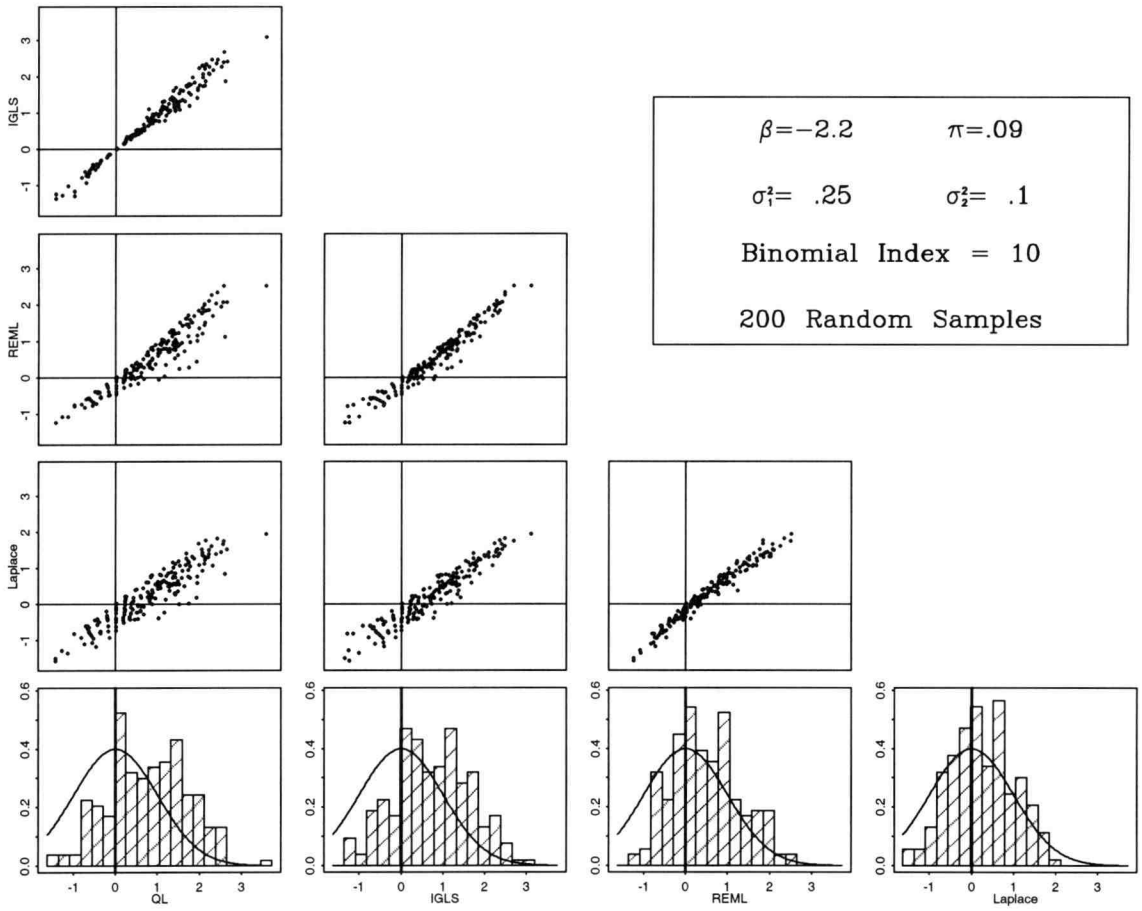


Table 29. Summary of Simulation Results for Cell Irradiation Data Set Scenario 4, Set 3

True β = -2.2, Binomial Index = 10, 200 samples (3^{rd} Set)

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.526	-1.583	-2.042	-2.050	.186	.150	.0573	-2.076	-2.025
IGLS	-2.526	-1.583	-2.042	-2.050	.186	.150	.0573	-2.076	-2.025
REML	-2.624	-1.617	-2.100	-2.098	.191	.102	.0467	-2.124	-2.072
Laplace	-2.715	-1.646	-2.148	-2.145	.200	.055	.0429	-2.173	-2.117

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-1.428	3.581	.811	.774	.945	.774	1.4923	.643	.905
IGLS	-1.348	3.099	.730	.730	.910	.730	1.3620	.604	.856
REML	-1.237	2.517	.365	.486	.816	.486	.9018	.373	.599
Laplace	-1.564	1.945	.216	.243	.744	.243	.6120	.140	.346

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	0	24	12.0 %	8.3 %	17.3 %
IGLS	0	20	10.0 %	6.6 %	15.0 %
REML	0	10	5.0 %	2.8 %	9.0 %
Laplace	0	0	0.0 %	0.0 %	1.8 %

Figure 26. Cell Irradiation Scenario 4, Sets 1, 2 & 3: β-estimates

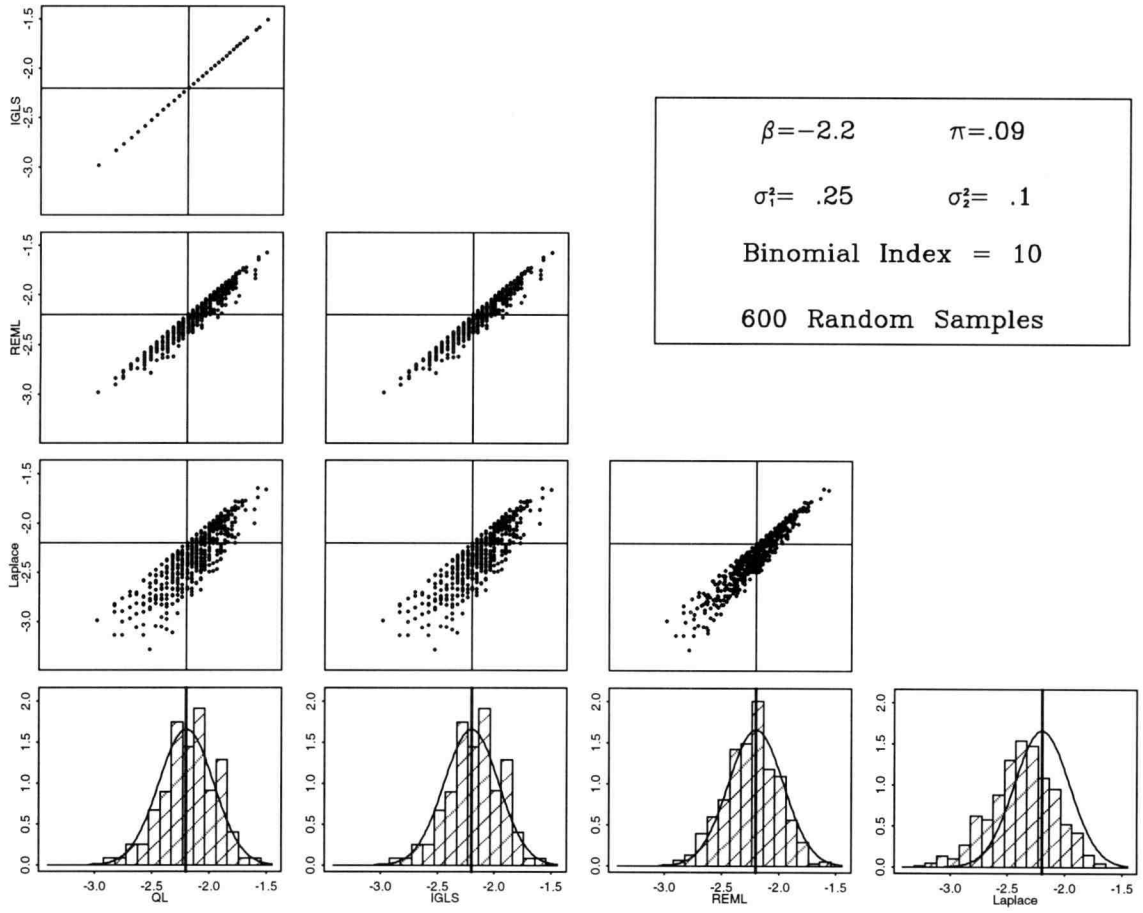


Figure 27. Cell Irradiation Scenario 4, Sets 1, 2 & 3: z-statistics for H_0 $\beta = -2.2$

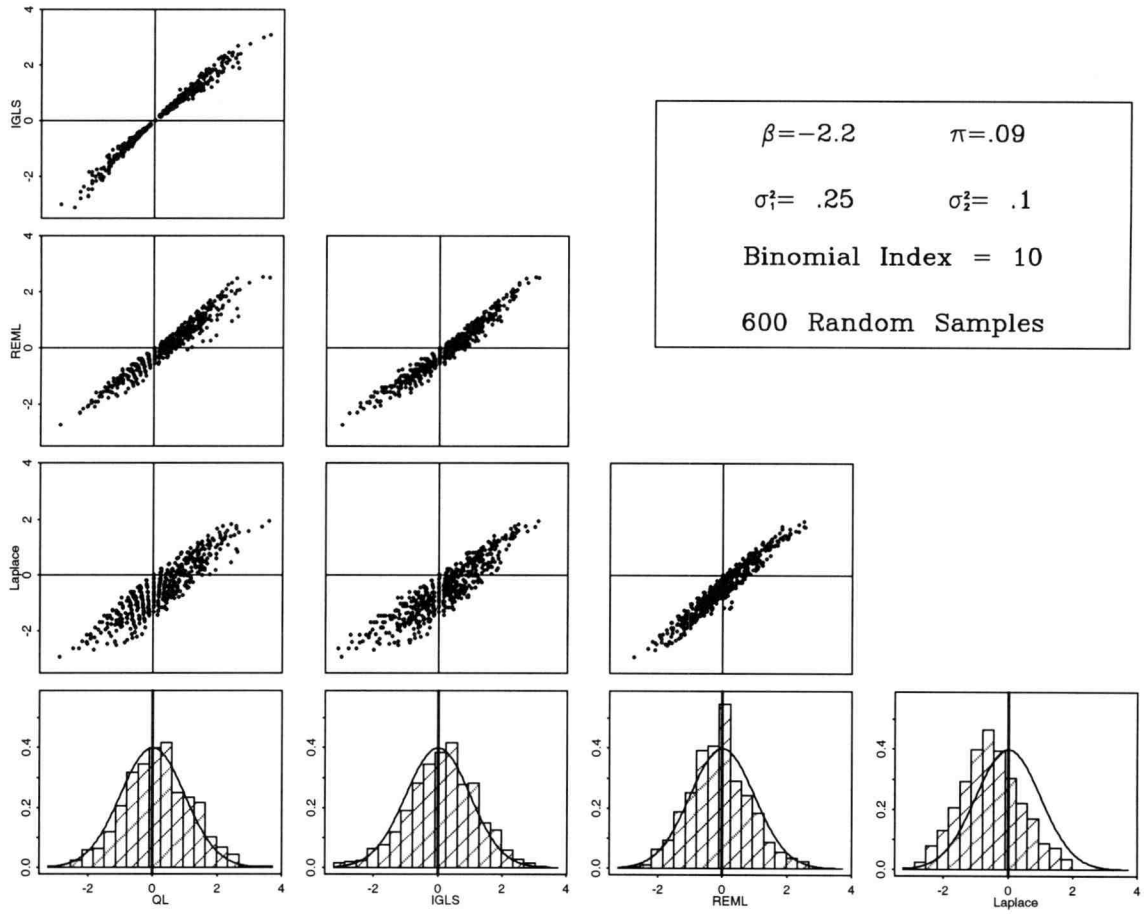


Table 30. Summary of Simulation Results for Cell Irradiation Data Set Scenario 4, Sets 1, 2 & 3 Combined

True β = -2.2, Binomial Index = 10, 600 samples (Combined)

Summary Measures of β Estimates

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.984	-1.506	-2.157	-2.175	.241	.025	.0587	-2.195	-2.156
IGLS	-2.984	-1.506	-2.157	-2.175	.241	.025	.0587	-2.195	-2.156
REML	-2.984	-1.571	-2.223	-2.239	.233	-.039	.0558	-2.258	-2.220
Laplace	-3.281	-1.646	-2.365	-2.369	.281	-.169	.1075	-2.392	-2.347

Summary Measures of z-scores

Method	min	max	med	avg	sd	bias	mse	95% Conf. Int. LCL	UCL
QL	-2.905	3.581	.186	.178	1.036	.178	1.1050	.095	.261
IGLS	-3.107	3.099	.192	.141	1.063	.141	1.1500	.056	.226
REML	-2.755	2.545	-.084	-.093	.908	-.093	.8331	-.166	-.020
Laplace	-2.923	1.945	-.605	-.539	.940	-.539	1.1741	-.614	-.464

More Summary Measures of z-scores

Method	# of z-scores <-1.96	>1.96	Type I Error Rate for 5% Level Test	95% Confidence Interval on Type I Error Rate LCL	UCL
QL	10	28	6.3 %	4.7 %	8.6 %
IGLS	20	23	7.2 %	5.4 %	9.5 %
REML	7	12	3.2 %	2.1 %	5.0 %
Laplace	38	0	6.3 %	4.7 %	8.6 %

5. CONCLUSIONS AND FINAL DISCUSSION

This thesis has presented research results regarding attempts to understand the usefulness of applying the Laplace integral approximation method for obtaining maximum likelihood estimates (MLEs) and likelihood ratio (LR) inferences in generalized linear models with two or more random effects. In Chapter 2, the background to the data problem was given, as well as information towards implementing three alternative modeling and estimation approaches to maximum likelihood (QL, IGLS, REML). Chapter 3 detailed the use of the Laplace integral approximation method for obtaining MLEs and LR inferences, as well as demonstrated for a few considered situations that this yielded very nearly exact MLEs and LR inferences. Chapter 4 then went on to empirically compare the estimates and inferences from Laplace with the alternative methods detailed in Chapter 2, using both real and simulated data sets. Here in Chapter 5, the results of the previous chapters are brought together to draw conclusions (§ 5.1) and discuss some unresolved issues (§ 5.2).

5.1 Conclusions

The results presented in the previous two chapters suggest that use of the Laplace approximation (along with a numerical optimization routine) obtains MLEs and LR inferences, that these are obtained much faster than if numerical integration had been used, and that it can be used for cases when numerical integration is infeasible. In Section 3.3.2, it was demonstrated that the shape of the log Laplace approximate likelihood is practically the same as the shape of the log-likelihood. Hence the parameters that maximize the log "Laplace likelihood" also maximize the log-likelihood and maximizing the profile log Laplace also maximizes the profile log-likelihood. The analyses results in Tables 14-18 demonstrate that obtaining these estimates and inferences is still time consuming due to the need for numerical optimization. Yet estimates and inferences were obtained in cases where numerical integration was infeasible (e.g., the Revertant Colony and Salamander Mating data sets) and much quicker even when numerical integration could be used (e.g., Cell Irradiation data set).

As evidenced in simulation scenarios 3 and 4 of Chapter 4, some further understanding near the boundary of the dispersion components needs to be addressed, both in the Laplace approximation's functional behavior and the optimization routine, as this was where some erratic behavior was observed. As mentioned in the text of Chapter 4, 2-node Gauss-Hermite (G-H) numerical integration was employed for parameter estimation and inferences but were not tabled nor plotted with the results of the simulations. In scenarios 1 and 2, the dispersion components' estimates between REML, Laplace and G-H were very similar and very few were 0 (i.e., never ran into a boundary problem). Yet in scenarios 3 and 4, there began to be departures among the three methods, with Laplace and G-H having several samples result in either dispersion component estimated as 0 while REML had very few estimates at or very near 0. Since the Nelder-Mead algorithm is not intended for use in a constrained problem, it is felt that the observed behavior is attributable more to it than towards the Laplace or G-H functional behavior. Therefore, some additional effort should be spent on understanding what the functional behavior is there and attempting to find an optimization routine that behaves well at the boundary.

The Laplace and REML methods gave superior results to those of QL and IGLS. The Laplace and REML estimates and inferences were quite similar, with some discrepancies seen in the Salamander Mating data and in simulation scenario 4. The naive modeling approach of QL gave estimates that were similar to those from IGLS, yet gave very poor inferences, as compared to others, under the assumption that model (1.1) is appropriate. In Chapter 2, it was demonstrated that IGLS was an approximation to REML. And in the simulations of Chapter 4, it was shown that IGLS performed worse than REML with regards to hypothesis testing. Breslow and Clayton (1993) demonstrate that REML is an approximation to Laplace. And the analyses on data sets, both real and simulated, suggest the approximation rather good as both gave similar results. There is no clear winner between Laplace and REML. REML gives much faster results and is easy to implement while Laplace obtains LR inferences as opposed to relying on asymptotic normality assumptions. The Laplace implementation (particularly the optimization routine) needs to be further looked into at the dispersion component boundary. Both methods need further investigation for other situations as the simulations here have focussed on the binomial nested random effects case.

All methods gave similar dispersion component estimates (with exception of Laplace in simulation set 4). Yet it is felt that inferences regarding the dispersion components are very questionable for these data sets and in this modeling framework.

5.2 Unresolved Issues

The most disturbing aspect of the results presented so far has been the apparent breakdown of the Laplace method when one or both dispersion component estimates are near the boundary. The fix to the Nelder-Mead algorithm for the boundary problem that was implemented for this research does not allow a dispersion component, once set to 0, to become positive again. Re-analyses of the data sets where the boundary problem appeared using an alternative optimization routine that is more adept for constrained parameters should be attempted to see whether or not it is a functional problem or an optimization problem. Alternative numerical optimization routines that have been shown to work well in constrained problems should be used for this.

The Nelder-Mead algorithm has three serious deficiencies for use as a numerical optimization routine for obtaining MLEs and LR inferences when model (1.1) holds. These are: the lack of speed, that it must be patched for boundary problems, and that there is a limit on number of parameters. The speed aspect of these three is the least serious as that can be overcome with faster machines. However the other two make for potential barriers for its use in many real data situations.

The occasional divergence of the Newton-Raphson routine for obtaining $\tilde{u}$ as needed for the Laplace approximation needs to be explored more and better understood. A very simplistic maximum step size patch was put into the routine which stopped all divergence problems. However, this may not be efficient code and may introduce other problems not yet encountered in the research presented here. If this method is to be implemented on a larger scale, some assurances on the Newton-Raphson convergence would be desirable.

The results presented here have been limited pretty much to binomial data and logit regression. There is a need to delve further into other regular

exponential families (e.g., Poisson, Exponential), non-canonical links (e.g., probit for binomial data), multiple predictor problems, and more than 2 random effects. The research has demonstrated that Laplace can be used for MLEs and LR inferences in a limited set of data problems. Further generalizations of the application need to be demonstrated as still producing good results. This will be quite time consuming and will need to be done with careful thought into what situations to pursue, as the optimization is quite time consuming.

A possible alternative to relying on numerical optimization for ML estimation and LR inferences would be the use of Laplace in the EM algorithm, as mentioned briefly in Section 2.2.2. Anderson and Aitkin (1985) suggest the use of EM along with numerical integration for obtaining MLEs. In Section 2.2.2, it was mentioned that Laplace could be used in the required integrals rather than numerical integration, which should reduce the computing time considerably. Yet there is still the issue of obtaining inferences without resorting to asymptotic normality assumptions, as LR inferences are preferable if obtainable. A suggestion for research is to see if the EM algorithm can be used here to find MLEs (for the nuisance parameters) in the reduced parameter space under the null hypothesis. If so, these EM computed restricted MLEs could be "plugged" into the Laplace approximate log-likelihood function to obtain profile log-likelihood plots or λ statistics. The thought is that the EM algorithm may converge to MLEs much faster than numerical optimization.

After the above issues have been explored and resolved, and a routine has been developed to obtain MLEs and LR inferences under the assumed model, then some exploration into both the robustness and model checking should be carried out. Exploring robustness should include, but not be limited to, non-normality of the random effects, such as heavy tailed but symmetric and skewed distributions. As for model checking, find some tools that are useful in assessing the adequacy of random effects being additive in the link, the normality of random effects, and the proper link function used.

An interesting observation from the analyses on the data sets and the simulations was the fact the MLEs of β were always more extreme than the other estimates, with REML estimates nearly as extreme. When looking at the different methods' estimates, they did seem to come from the two camps: QL/IGLS and REML/Laplace. Two interesting issues arise as to what are the $\hat{\beta}$

estimating for the different methods (IGLS, REML, Laplace) and if this is related to the issue of population averaged vs. subject specific effects mentioned in Breslow and Clayton (1993). If they are intended to be estimating the same thing (as they were each derived from a common underlying model), then one group must be biased. Two additional questions that arise are what does $g^{-1}(X^{\mathrm{T}}\hat{\beta})$ estimate for the different models and how does one estimate means, μ, on the original scale? These need to be better understood so as to correctly interpret results.

5.3 Summary Remarks

Much work in recent years has been spent by several researchers worldwide attempting to address the modeling, estimation and inference aspects of categorical data collected with more than one random effect. The research presented here builds on that by Liu and Pierce (1993) which looked at using the Laplace method for a single random effect. Although REML-like methods have been suggested by others, little or no effort was placed on demonstrating the adequacy of asymptotic-based inferences. Here, each methods' behavior regarding inferences has been a major focus in comparisons, and the conclusion is that REML appears to do a fairly good job.

However, much work is still needed beyond the results presented here. The implementation of ML via Laplace used herein has some inadequacies that should be resolved before being used broadly. There is a need to generalize the results more and understand how model assumption violations affect the results and be detected.

The research here has been an attempt to obtain estimates and inferences under the framework of maximum likelihood, hoping that in doing so the theory behind LR inferences will hold. This, I believe, has indeed been demonstrated. With the Laplace approximate likelihood function, the more desirable likelihood based results are obtainable with a reasonable amount of computing time and effort.

BIBLIOGRAPHY

Abramowitz, M. and Stegun I. A. (1972) *Handbook of Mathematical Functions.* New York: Dover.

Aitkin, M., Anderson, D. A., Francis, B., and Hinde, J. (1989) *Statistical Modeling in GLIM.* New York: Oxford University Press.

Anderson, D. A. and Aitkin, M. (1985) Variance Component Models with Binary Response: Interviewer Variability. *J. R. Statist. Soc.* B, **47**, 203–210.

Barndorff-Nielsen, O. E. and Cox, D. R. (1989) *Asymptotic Techniques for Use in Statistics.* London: Chapman and Hall.

Breslow, N. E. (1984) Extra–Poisson Variation in Log–linear Models. *Appl. Statist.*, **33**, 38–44.

Breslow, N. E. and Clayton, D. G. (1993) Approximate Inference in Generalized Linear Mixed Models. *JASA*, **88**, 9-25.

Carroll, R. J. and Rupert, D. (1982) Robust Estimation in Heteroscedastic Linear Models. *Annals of Statistics*, **10**, 429–441.

Cox, D. R., Hinkley, D. V., Reid, N., and Snell, E. J. (1991) *Statistical Theory and Modeling: in Honour of Sir David Cox, FRS.* London: Chapman and Hall.

Cox, D. R. and Reid, N. (1987) Parameter Orthogonality and Approximate Conditional Inference. *J. R. Statist. Soc.* B, **49**, 1–39.

Davison, A. C. (1986) Approximate Predictive Likelihood. *Biometrika*, **73**, 323–332.

DeBruijn, N. G. (1961) *Asymptotic Methods in Analysis*, 2nd ed. Amsterdam: North-Holland.

Dempster, A. P., Laird, N. M., and Rubin, D. B. (1977) Maximum Likelihood from Incomplete Data via the *EM* Algorithm. *J. R. Statist. Soc.* B, **39**, 1–22.

Fellner, W. H. (1986) Robust Estimation of Variance Components. *Technometrics*, **28**, 51–60.

Fellner, W. H. (1987) Sparse Matrices, and the Estimation of Variance Components by Likelihood Methods. *Communications in Statistics* B, **16**, 439–463.

Firth, D. and Harris, I. R. (1991) Quasi–likelihood for Multiplicative Random Effects. *Biometrika*, **78**, 545–555.

BIBLIOGRAPHY (Continued)

Gilmour, A. R., Anderson, R. D., and Rae, A. L. (1985) The Analysis of Binomial data by a Generalized Linear Mixed Model. *Biometrika*, **72**, 593–599.

Goldstein, H. (1991) Nonlinear Multilevel Models, with an Application to Discrete Response Data. *Biometrika*, **78**, 44–51.

Green, P. J. (1987) Penalized Likelihood for General Semi-Parametric Regression Models. *International Statistical Review*, **55**, 245–259.

Harville, D. A. (1977) Maximum Likelihood Approaches to Variance Component Estimation and to Related Problems. *JASA*, **72**, 320–340.

Haseman, J. K. and Kupper, L. L. (1979) Analysis of Dichotomous Response Data from Certain Toxilogical Experiments. *Biometrics*, **35**, 281–293.

Karim, M. R. and Zeger, S. L. (1992) Generalized Linear Models with Random Effects; Salamander Mating Revisited. *Biometrics*, **48**, 631–644.

Kass, R. E., Tierney, L., and Kadane, J. B. (1990) Asymptotic Evaluation of Integrals Arising in Bayesian Inference. Technical Report # 502. Department of Statistics, Carnegie Mellon University.

Kotz, S., Johnson, N. L., and Read, C. B. (1985) *Encyclopedia of Statistical Sciences*. New York: John Wiley & Sons, Inc.

Liang, K. Y. and Zeger, S. L. (1986) Longitudinal Data Analysis Using Generalized Linear Models. *Biometrika*, **73**, 13–22.

Liu, Q. and Pierce, D. A. (1993) Heterogeneity in the Mantel-Haenszel-Type Models. *Biometrika*, **80**, 543-556.

The MathWorks, Inc. (1990) *Optimization Toolbox User's Guide*, Natick, MA: The MathWorks, Inc.

McCullagh, P. (1983) Quasi-likelihood Functions. *Annals of Statistics*, **11**, 59–67.

McCullagh, P. and Nelder, J. A. (1989) *Generalized Linear Models*, 2nd ed. London: Chapman and Hall.

McGilchrist, C. A. and Aisbett, C. W. (1991) Restricted BLUP for Mixed Linear Models. *Biometrical Journal*, **33**, 131-141.

Mood, A. M., Graybill, F. A., and Boes, D. C. (1974) *Introduction to the Theory of Statistics*, 3rd ed. New York: McGraw-Hill.

Morton, R. (1987) A Generalized Linear Model with Nested Strata of Extra–Poisson Variation. *Biometrika*, **74**, 247–257.

BIBLIOGRAPHY (Continued)

Nelder, J. A. and Mead, R. (1965) A Simplex Method for Function Minimization, *Computer Journal*, **7**, 308–313.

Patterson, H. D. and Thompson, R. (1971) Recovery of Inter-block Information when Block Sizes are Unequal. *Biometrika*, **58**, 545–554.

Pierce, D. A. and Sands, B. R. (1975) Extra–Bernoulli Variation in Regression of Binary Data. Technical Report # 46. Department of Statistics, Oregon State University.

Powell, M. J. D. (1978) A Fast Algorithm for Nonlineary Constrained Optimization Calculations, *Numerical Analysis*, ed. G.A. Watson, Lecture Notes in Mathematics, Springer Varleg, Vol. 630.

Preisler, H. K. (1989) Analysis of a Toxicological Experiment Using a Generalized Linear Model with Nested Random Effects. *International Statistical Review*, **57**, 145–159.

Prentice, R. L. (1988) Correlated Binary Regression with Covariates Specific to Each Binary Observation. *Biometrics*, **44**, 1033–1048.

Schall, R. (1991) Estimation in Generalized Linear Models with Random Effects. *Biometrika*, **78**, 719–727.

Searle, S. R., Casella, G., and McCulloch, C. E. (1992) *Variance Components.* New York: J. Wiley and Sons.

Shanno, D. F. (1970) *Conditioning of Quasi-Newton Methods for Function Minimization*, Mathematics of Comput., **24**, 647-656.

Simpson, D. G. and Margolin, B. H. (1986) Recursive Nonparametric Testing for Dose–response Relationships Subject to Downturns at High Doses. *Biometrika*, **73**, 589–596.

Stiratelli, R., Laird, N., and Ware, J. H. (1984) Random–Effects Models for Serial Observations with Binary Response. *Biometrics*, **40**, 961–971.

Thisted, R. A. (1988) *Elements of Statistical Computing.* New York: Chapman and Hall.

Tierney, L. and Kadane, J. B. (1986) Accurate Approximations for Posterior Moments and Marginal Densities. *JASA*, **81**, 82–86.

Tierney, L., Kass, R. E. and Kadane, J. B. (1989) Fully Exponential Laplace Approximations to Expectations and Variances of Nonpositive Functions. *JASA*, **84**, 710–716.

Wedderburn, R. W. M. (1974) Quasilikelihood Functions, Generalized Linear Models and the Gauss-Newton Method. *Biometrika*, **61**, 439–447.

BIBLIOGRAPHY (Continued)

Williams, D. A. (1982) Extra–binomial Variation in Logistic Linear Models. *Appl. Statist.*, **31**, 144–148.

Wong, G. Y. and Mason, W. M. (1985) The Hierarchical Logistic Regression Model for Multilevel Analysis. *JASA*, **80**, 513–524.

Zeger, S. L. and Karim, M. R. (1991) Generalized Linear Models with Random Effects; A Gibbs Sampling Approach. *JASA*, **86**, 79–86.

Zeger, S. L. and Liang, K.–Y. (1986) Longitudinal Data Analysis for Discrete and Continuous Outcomes. *Biometrics*, **42**, 121–130.

APPENDICES

A. PROGRAMS FOR DISTRIBUTION SPECIFIC CALCULATIONS

B1.m

```
function [f]= B1(theta,m,dist)
     % B1 calculates the function b(theta) involved in the regular
     % exponential family notation of the model.  theta can either be
     % a (q-by-1) vector or a (q-by-k) matrix for k different
     % realizations.  m is a vector of binomial indeces if the data
     % are binomial counts or of 1s if Poisson counts. dist indicates
     % whether the model assumes Poisson (dist=1) or binomial
     % (dist=2) counts.
%    James Pratt, 1-10-91
%    Dept. of Statistics
%    Oregon State University
f=exp(theta);
if dist>1,
   f=log(1+f).*m;
   end
end;
```

Bw.m

```
function  [g,h]= Bw(theta,m,dist)
     % Bw calculates the function b(theta) involved in the regular
     % exponential family notation of the model.  theta is a (q-by-1)
     % vector.  m is a vector of binomial indeces if the data are
     % binomial counts or of 1s if Poisson counts.  dist indicates
     % whether the model assumes Poisson (dist=1) or binomial
     % (dist=2) counts.  The function returns the first derivative of
     % b with respect to theta (as g) and the second derivative of b
     % with respect to theta (as h).
%    James Pratt, 1-10-91
%    Dept. of Statistics
```

```
%   Oregon State University
B=exp(theta);
if dist==1,
  g=B;h=B;
else
   g=B./(1+B);
   g=g.*m;
   h=g./(1+B);
end;
```

link.m

```
function  [g,h]= link(mu,m,dist)
     % Is passed mu, the current estimate of E(y) where y is a vector
     % of counts.  m is a vector of binomial indeces if the data are
     % binomial counts or of 1s if Poisson counts.  dist indicates
     % whether the model assumes Poisson (dist=1) or binomial
     % (dist=2) counts.  It returns two column vectors:
     % g(mu) and 1/g'(mu), where g is the canonical link function
     % g(mu)=theta.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
mu=mu+.5*(mu==0);  % ensure don't take log of 0.
if dist==1,
  g=log(mu);
  h=mu;
else
  mu=mu-.5*(mu==m);  % ensure m-mu is positive.
  g=log(mu)-log(m-mu);
  h=mu.*(m-mu)./m;
end
```

B. PROGRAMS FOR QL, IGLS AND REML ROUTINES

initial.m

```
% Matlab script file: initial.m     updated: Sept 8, 1992
% Finds initial beta estimates ignoring the random effects as in
% GLIM.  It also finds sigma squared assuming a simple
% overdispersion model also as in GLIM.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
begin=clock;
[z W]=link(y,m,dist);
W=diag(W);
betahi=(x'*W*x)\(x'*W*z);
  theta=x*betahi;
  [mu Vmu]=Bw(theta,m,dist);
  W=diag(Vmu);
  z=theta+(W)\(y-mu);
  beta0=betahi;
  betahi=(x'*W*x)\(x'*W*z);
while max(abs(beta0-betahi)./abs(beta0))>.01,
  theta=x*betahi;
  [mu Vmu]=Bw(theta,m,dist);
  W=diag(Vmu);
  z=theta+(W)\(y-mu);
  beta0=betahi;
  betahi=(x'*W*x)\(x'*W*z);
end;
[mu Vmu]=Bw(x*betahi,m,dist);
W=diag(Vmu);
oversig=(y-mu)'*inv(W)*(y-mu)/(N-q);
covbetahi=oversig*inv(x'*W*x);
[betahi diag(covbetahi)]
```

```
oversig
timeinit=etime(clock,begin)
```

gls.m

```
     % Matlab function gls.m       updated: Sep 8, 1992
function [betah, disph]=gls(y,m,x,betah,A1,A2,a,dist)
     % Using a current estimate of the fixed effects (betah), find
     % their generalized least squares estimates using Williams'
     % model III approach with an obvious extension to two random
     % effects.  Also, find estimates of the dispersion components
     % (disph) using method of moments and McCullagh & Nelder's
     % "natural" quadratic forms.
%    James Pratt, 1-10-91
%    Dept. of Statistics
%    Oregon State University
begin=clock;
P2=A2*A2';
P1=A1*A1';
[mu Vmu]=Bw(x*betah,m,dist);
R=y-mu;
d=sum(Vmu);
Q1=R'*P1*R-d;
Q2=R'*P2*R-d;
W=diag(Vmu);
el1=trace(P1*W*P1*W);
el2=trace(P2*W*P2*W);
el12=trace(P1*W*P2*W);
el21=trace(P2*W*P1*W);
disph=inv([[el1 el21]' [el12 el2]'])*[Q1 Q2]'
if min(disph)<0,
  if disph(1)<0,
    disph(1)=0;
    disph(2)=(Q2/el2)*(Q2>0)+(Q1/el12)*(Q2<0);
  else
```

```
    disph(2)=0;
    disph(1)=(Q1/el1)*(Q1>0)+(Q2/el21)*(Q1<0);
  end
end;
D=(P1.*disph(1))+(P2.*disph(2));
disp0=[0 0]';
iter=1;
while (max(abs(disph-disp0)./abs(disp0+.00000001))>.01) & (iter<8),
   iter=iter+1;
   theta=x*betah;
   [mu Vmu]=Bw(theta,m,dist);
   W=diag(Vmu);
   z=theta+(W)\(y-mu);
   W=inv(inv(W)+D);
   beta0=betah;
   betah=(x'*W*x)\(x'*W*z);
   while max(abs(beta0-betah)./abs(beta0))>.01,
     theta=x*betah;
     [mu Vmu]=Bw(theta,m,dist);
     W=diag(Vmu);
     z=theta+(W)\(y-mu);
     W=inv(inv(W)+D);
     beta0=betah;
     betah=(x'*W*x)\(x'*W*z);
   end;
   disp0=disph;
   [mu Vmu]=Bw(x*betah,m,dist);
   R=y-mu;
   d=sum(Vmu);
   Q1=R'*P1*R-d;
   Q2=R'*P2*R-d;
   W=diag(Vmu);
   el1=trace(P1*W*P1*W);
   el2=trace(P2*W*P2*W);
   el12=trace(P1*W*P2*W);
```

```
    el21=trace(P2*W*P1*W);
    disph=inv([[el1 el21]' [el12 el2]'])*[Q1 Q2]';
    if min(disph)<0,
      if disph(1)<0,
        disph(1)=0;
        disph(2)=(Q2/el2)*(Q2>0)+(Q1/el12)*(Q2<0);
      else,
        disph(2)=0;
        disph(1)=(Q1/el1)*(Q1>0)+(Q2/el21)*(Q1>0);
      end
    end
    D=(P1.*disph(1))+(P2.*disph(2));
end;
if iter>7,
  disp('exceed maximum iteration steps in gls.m')
  pause
end;
timegls=etime(clock,begin);
disp('IGLS Completed')
end;
```

reml.m

```
    % Matlab function reml.m       updated: Sept 7, 1992
function [beta, disph, mode]= ...
  reml(y,m,x,betah,A,a,dispgls,modegls,dist,maxit)
    % Using a current estimate of the fixed effects (betah), find
    % their REML estimates, along with the REML random effects
    % estimates (mode) by treating the linearized model as a normal
    % linear mixed model and using Harville's (1977) method for
    % estimation.  See Schall (1991) and Breslow & Clayton (1992)
    % for details.
%   James Pratt, 1-10-91
%   Dept. of Statistics
```

```
%   Oregon State University
begin=clock;
theta=x*betah+A*modegls;
[mu Vmu]=Bw(theta,m,dist);
W=diag(Vmu);
iW=inv(W);
q=length(betah);
D=diag([ones(1,a(1))*dispgls(1) ones(1,a(2))*dispgls(2)]);
z=theta+iW*(y-mu);
Var=inv(iW+A*D*A');
beta=inv(x'*Var*x)*x'*Var*z;
mode=D*A'*Var*(z-x*beta);
CprimeC=[[x'*W*x x'*W*A]' [A'*W*x A'*W*A+inv(D)]'];
invCC=inv(CprimeC);
T=invCC(q+1:q+sum(a),q+1:q+sum(a));
v1=trace(T(1:a(1),1:a(1)));den1=a(1)-v1/dispgls(1);
v2=trace(T)-v1;den2=a(2)-v2/dispgls(2);
disph(1)=mode(1:a(1))'*mode(1:a(1))/den1;
disph(2)=mode(a(1)+1:sum(a))'*mode(a(1)+1:sum(a))/den2;
disp0=dispgls';
beta0=betah;
iter=1;
while (max(abs(disph-disp0)./abs(disp0+.00000001))>.01) &
(iter<(maxit+1)) ...
      & max(abs(beta0-beta)./abs(beta0))>.01,
   iter=iter+1;
   theta=x*beta+A*mode;
   [mu Vmu]=Bw(theta,m,dist);
   W=diag(Vmu);
   iW=inv(W);
   D=diag([ones(1,a(1))*disph(1) ones(1,a(2))*disph(2)]);
   z=theta+iW*(y-mu);
   Var=inv(iW+A*D*A');
   beta0=beta;
   disp0=disph;
```

```
    beta=inv(x'*Var*x)*x'*Var*z;
    mode=D*A'*Var*(z-x*beta);
CprimeC=[[x'*W*x x'*W*A]' [A'*W*x A'*W*A+inv(D)]'];
invCC=inv(CprimeC);
T=invCC(q+1:q+sum(a),q+1:q+sum(a));
    v1=trace(T(1:a(1),1:a(1))); den1=a(1)-v1/disph(1);
    v2=trace(T)-v1; den2=a(2)-v2/disph(2);
disph(1)=mode(1:a(1))'*mode(1:a(1))/den1;
disph(2)=mode(a(1)+1:sum(a))'*mode(a(1)+1:sum(a))/den2;
end;
if iter>maxit,
  disp('exceed maximum iteration steps in reml.m')
end;
timereml=etime(clock,begin);
disp('REML Completed')
end;
```

C. LOG-LIKELIHOOD APPROXIMATION ROUTINES

modes.m

```
     % Matlab function modes.m
function wnew = modes(y,m,theta,A,D,dist,steps)
     % modes calculates the mode of the log of the integrand
     % of the integral.  The mode is found using Newton-Raphson
     % with at most steps iterations. x is the fixed effects
     % design matrix, beta is the current estimate of the fixed
     % effects, A is the random effects design matrix, wold is
     % the current guess at the mode, D is the diagonal vector of
     % the random effects covariance matrix, and dist determines
     % the assumed conditional distribution.
%    James Pratt, 1-10-91
%    Dept. of Statistics
%    Oregon State University
iter=1;
D=diag(D);
ID=inv(D);
I=eye(D);
[mu V]=Bw(theta,m,dist);
R=y-mu;
V=diag(V);
wold=inv(ID+A'*V*A)*A'*R;
wnew=wold+inv(A'*V*A+D)*(A'*R-D*wold);
check=1;
while check & max(abs(A'*R-D*wnew))>1e-4 & iter<steps,
   iter=iter+1;
   wold=wnew;
   [mu V]=Bw(theta+A*wold,m,dist);
   R=y-mu;
   V=diag(V);
   wnew=wold+inv(A'*V*A+D)*(A'*R-D*wold);
```

```
   check=max(abs(wnew-wold)./abs(wold+.000001))> .001;
end;
if iter>=steps,
  disp('exceed steps in modes.m')
end;
```

Laplace.m

```
     % Matlab function Laplace.m     updated: March 28, 1992
function [int,w0,L0]=Laplace(y,m,theta,A,D,a,disp,dist)
     % Laplace computes the log-Laplace approximation to the
     % likelihood function at the passed values of beta and
     % dispersion components.  It finds w0, the mode of the log of
     % the integrand and returns it.  It also finds A0, the negative
     % of the Hessian of the log of the integrand evaluated at w0.
     % It finds and retuns L0, the Cholesky decomposition of A0.
     % w0 and L0 are returned for use in Gauss-Hermite integration,
     % which also calls Laplace.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
w0=modes(y,m,theta,A,D,dist,40);
[mu V]=Bw(theta+A*w0,m,dist);
A0=A'*diag(V)*A+diag(D);

%  Find L0, the lower triangular square root of A0
L0=chol(A0)';

%  compute int, the log-Laplace approx. to the integral.
eta=theta+A*w0;
b=B1(eta,m,dist);
int=y'*eta-sum(b)-.5*w0'*diag(D)*w0-sum(log(diag(L0)))-
.5*a*log(disp');
end;
```

GHwts.m

```
% Matlab script file GHwts.m
% Creates several variables containing Gauss-Hermite nodes and
% weights for 1, 2, ... 10, 12, 16, and 20 node quadrature.
% Values are taken from Abramawitz and Stegum.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
n1=0;
w1=1;

n2=[.7071067811 -.7071067811]';
w2=[.88622692545 .88622692545]';
w2=w2/sqrt(pi);

n3=[1.2247448714 0 -1.22474488714]';
w3=[.29540897515 1.1816359006 .2954097515]';
w3=w3/sqrt(pi);

n4=[1.6506801239 .524647633 -.5246476233 -1.6506801239]';
w4=[.081312835447 .80491409001 .80491409001 .081312835447]';
w4=w4/sqrt(pi);

n5=[2.0201828705 .9585724646 0 -.958572646 -2.0201828705]';
w5=[.019953242059 .39361932315 .94530872048 .3961932315 ...
    .019953242059]';
w5=w5/sqrt(pi);

n6=[2.3506049737 1.3358490740 .4360774119 ...
    -.4360774119 -1.3358490740 -2.3506049737]';
w6=[4.5300099055e-3 .15706732032 .72462959522 ...
    .72462959522 .15706732032 4.5300099055e-3]';
w6=w6/sqrt(pi);
```

```
n7=[2.6519613568 1.6735516288 .8162878829 0 ...
    -.8162878829 -1.6735516288 -2.651613568]';
w7=[9.7178124509e-4 5.4515582818e-2 4.2560725261e-1 ...
    8.1026461755e-1 4.2560725261e-1 5.4515582818e-2 ...
    9.7178124509e-4]';
w7=w7/sqrt(pi);

n8=[2.9306374203 1.9816567567 1.1571937125 .3811869902 ...
    -.3811869902 -1.1571937125 -1.9816567567 -2.9306374203]';
w8=[1.9960407221e-4 .017077983007 .20780232582 .66114701256 ...
   .66114701256 .20780232582 .017077983007 1.9960407221e-4]';
w8=w8/sqrt(pi);

n9=[3.1909932018 2.2665805845 1.4685532892 .7235510188 0 ...
    -.7235510188 -1.4685532892 -2.2665805845 -3.1909932018]';
w9=[3.9606977263e-5 4.9436242755e-3 8.8474527394e-2 4.32651559e-1 ...
    7.2023521561e-1 4.32651559e-1 8.8474527394e-2 4.9436242755e-3 ...
    3.9606977263e-5]';
w9=w9/sqrt(pi);

n10=[3.4361591188 2.5327316742 1.7566836493 1.0366108298 ...
    .3429013272 -.3429013272 -1.0366108298 -1.7566836493 ...
    -2.5327316742 -3.4361591188]';
w10=[7.6404328552e-6 1.3436457468e-3 .033874394456 ...
    .24013861108 .61086263374 .61086263374 .24013861108 ...
    .033874394456 1.3436457468e-3 7.6404328552e-6]';
w10=w10/sqrt(pi);

n12=[3.8897248979 3.0206370251 2.2795070805 1.5976826352 ...
    .9477883912 .3142403763 -.3142403763 -.9477883912 ...
   -1.5976826352 -2.2795070805 -3.0206370251 ...
   -3.8897248979]';
w12=[2.76585516844e-7 8.5736870436e-5 3.9053905846e-3 ...
    .051607985616 .26049231026 .57013523626 .57013523626 ...
    .26049231026 .051607985616 3.905390584e-3 ...
```

```
    8.5736870436e-5 2.76585516844e-7]';
w12=w12/sqrt(pi);

n16=[4.6887389393 3.8694479049 3.176999162 2.5462021578 ...
     1.9517879909 1.3802585392 .8229514491 .2734810461 ...
     -.2734810461 -.8229514491 -1.3802585392  -1.9517879909 ...
     -2.5462021578 -3.176999162 -3.8694479049 -4.6887389393 ]';

w16a=[2.654807474e-10 2.3209808449e-7 2.7118600925e-5 ...
      9.3228400862e-4 1.2880311536e-2 8.3810041399e-2 ...
      2.8064745853e-1 5.0792947902e-1];
w16b=[5.0792947902e-1 2.8064745853e-1 8.3810041399e-2 ...
      1.2880311536e-2 9.3228400862e-4 2.7118600925e-5 ...
      2.3209808449e-7 2.654807474e-10];
w16=[w16a w16b]';
w16=w16/sqrt(pi);

n20a=[5.3874808900 4.6036824496 3.9447640401 3.3478545674 ...
    2.7888060584 2.2549740021 1.7385377121 1.2340762454];
n20b=[.7374737286 .2453407083 -.2453407083 -.7374737286 ...
   -1.2340762454 -1.7385377121 -2.2549740021 -2.7888060584];
n20c=[-3.3478545674 -3.9447640401 -4.6036824496 -5.3874808900];
n20=[n20a n20b n20c]';
w20=[2.2293936455e-13 4.3993409923e-10 1.0860693707e-7 ...
     7.8025564785e-6 2.2833863602e-4 3.2437733422e-3];
w20=[w20 .024810520888 .10901720602 .28667550536 .46224366960 ...
     .46224366960 .28667550536 .10901720602 .024810520888];
w20=[w20 3.2437733422e-3 2.2833863602e-4 7.8025564785e-6 ...
     1.0860693707e-7 4.3993409923e-10 2.2293936455e-13]';
w20=w20/sqrt(pi);
```

GHint.m

```
     % Matlab function GHint.m        updated March 28, 1992
function [int,mult]=GHint(y,m,theta,A,D,a,disp,dist,M,weights,nodes)
     % GHint performs Gauss-Hermite "directed" numerical integration.
     % M is the number of nodes each dimension will use.  weights and
     % nodes are vectors of length M.  The Laplace function is called
     % to get Laplace factor as well as w0 and L0.  It returns the
     % M-node G-H integral approximation.
%    James Pratt, 1-10-91
%    Dept. of Statistics
%    Oregon State University
time1=clock;
%  Call Laplace for multiplier, w0 and L0.
[mult w0 L0]=Laplace(y,m,theta,A,D,a,disp,dist);
etime(clock,time1)
time2=clock;
int=zeros(1,a(1));
check=length(a)-1;
if check,
  index=ones((a(2)/a(1))+1,1);
  z=nodes(1)*[eye(a(1)) kron(eye(a(1)),ones(1,a(2)/a(1))) ]';
  wts=(weights(1)-1)*(z~=0) + ones(z);
  mode=kron(w0,ones(1,a(1))).*(z~=0);
else,
  index=1;
  z=nodes(1)*eye(a(1));
  wts=(weights(1)-1)*eye(a(1))+1;
  mode=diag(w0);
end;
D=diag(D);
[mu V]=Bw(theta+A*w0,m,dist);
b=B1(theta+A*w0,m,dist);
A1=A(:,1:a(1));
hw0=y'*A*mode - b'*A1 - .5*diag(mode'*D*mode)';
```

```
V0=A'*diag(V)*A + D;
L0=sqrt(2)*inv(L0');
go=1;
while go,
   i=length(index);
   J=i-1;
   w=mode+L0*z;
   b=B1(theta+A*sum(w')',m,dist);
   hw=y'*A*w - b'*A1 - .5*diag(w'*D*w)';
   int=int+exp(hw-hw0+.5*diag((L0*z)'*V0*L0*z)'+sum(log(wts)));
   while i>1,
       if index(i)==M,
         index(i)=1;
         newn=nodes(1);
         newwt=weights(1);
         for j=1:a(1),
           z(a(1)+(j-1)*J+i-1,j)=newn;
           wts(a(1)+(j-1)*J+i-1,j)=newwt;
         end;
         i=i-1;
       else,
         index(i)=index(i)+1;
         newn=nodes(index(i));
         newwt=weights(index(i));
         for j=1:a(1),
           z(a(1)+(j-1)*J+i-1,j)=newn;
           wts(a(1)+(j-1)*J+i-1,j)=newwt;
         end;
         i=-1;
       end;
   end;
   if i==1,
     if index(1)==M,
       go=0;
     else,
```

```
        index(1)=index(1)+1;
        z(1:a(1),:)=nodes(index(1))*eye(a(1));
        wts(1:a(1),:)=(weights(index(1))-1)*eye(a(1))+1;
      end;
    end;
end;
etime(clock,time2)
int=sum(log(int));
end;
```

D. NUMERICAL OPTIMIZATION ROUTINES

fminjim.m

```
     % Matlab funtion fminjim.m      updated: March 28,1992
function [x, cnt] = ...
    fminjim(funfcn,x,prnt,tol,tol2,P1,P2,P3,P4,P5,P6,P7,P8,P9)
     % fminjim finds the minimum of a function of several variables.
     % X=fminjim('FUN',X0) starts at the matrix X0 and finds a
     % minimum to the function which is described in FUN (usually an
     % M-file: FUN.M).  The function 'FUN' should return a scalar
     % function value: F=FUN(X).
     % X=fminjim('FUN',X0,prnt,tol,tol2) allows optional parameters
     % to be defined. prnt controls how much display output is given;
     % set to 1 for a tabular display of results, (default is no
     % display: 0). If set >1 then will cause print only at the
     % number of iteration steps.
     % tol is a measure of the precision required for the values of
     % x at the solution. tol2 is a measure of the precision
     % required of the objective function at the solution.
     % X=fminjim('FUN',X0,prnt,tol,tol2,P1,P2,P3,P4,...) allows
     % variables, P1, P2, P3, P4 ,...to be passed directly to FUN:
     % [F,G]=FUN(X,P1,P2,P3,P4). More can be added.
     % fminjim uses a Simplex search method.
% C. Moler, 8-19-86
% Revised Andy Grace, 6-22-90
% Copyright (c) 1986-88 by the MathWorks, Inc.
% Tailored from FMINS by James Pratt, 7-11-91 for dissertation
% research purposes.
% Ref: D. J. Woods, Report 85-5, Dept. Math. Sciences, Rice Univ.,
% May, 1985.
if nargin<3, prnt=0; end
if nargin<4, tol=1.e-3; end
if nargin<5, tol2=.01; end
```

```
evalstr = [funfcn];
if ~any(funfcn<48)
        evalstr=[evalstr, '(x'];
        for i=1:nargin - 5
                evalstr = [evalstr,',P',num2str(i)];
        end
        evalstr = [evalstr, ')'];
end

n = prod(size(x));
ndisp=sum(P6>0);
maxit = 200*n; % change to 200*n.

% Set up a simplex near the initial guess.
xin = x(:);
v = 0.9*xin;
x(:) = v; fv = eval(evalstr);
for j = 1:n
   y = xin;
   if y(j) ~= 0
      y(j) = 1.1*y(j);
   else
      y(j) = 0.1;
   end
   v = [v y];
   x(:) = y; f = eval(evalstr);
   fv = [fv  f];
end
[fv,j] = sort(fv);
v = v(:,j);

cnt = n+1;
prntstep=prnt;
oldcnt=cnt;
if prnt
```

```
    clc
    format compact
    format short e
    home
    cnt
    disp('initial ')
    disp(' ')
    v
    f
end

alpha = 1;  beta = 1/2;  gamma = 2;
[n,np1] = size(v);
onesn = ones(1,n);
ot = 2:n+1;
on = 1:n;

% Iterate until the diameter of the simplex is less than tol.
while cnt < maxit
if max(max(abs(v(:,ot)-v(:,onesn)))) <= tol &  ...
        max(abs(fv(1)-fv(ot))./abs(fv(1))) <= tol2, break, end
% added code to patch for boundary problem with dispersion components
        zed=max(v');
        if abs(zed(n))<1.e-5,
          v(n,1)=-1;
          break,
        end
        if (abs(zed(n-1))<1.e-5) & ndisp==2,
          v(n-1,1)=-1;
          break,
        end
% end of added code
% One step of the Nelder-Mead simplex algorithm

vbar = (sum(v(:,on)')/n)'; % Mean value
```

```
vr = (1 + alpha)*vbar - alpha*v(:,n+1);
x(:) = vr; fr = eval(evalstr);
cnt = cnt + 1;
vk = vr;  fk = fr; how = 'reflect ';
if fr < fv(n)
   if fr < fv(1)
      ve = gamma*vr + (1-gamma)*vbar;
      x(:) = ve; fe = eval(evalstr);
   cnt = cnt + 1;
      if fe < fv(1)
         vk = ve; fk = fe;
         how = 'expand  ';
      end
   end
else
   vt = v(:,n+1); ft = fv(n+1);
   if fr < ft
      vt = vr;  ft = fr;
   end
   vc = beta*vt + (1-beta)*vbar;
   x(:) = vc; fc = eval(evalstr);
   cnt = cnt + 1;
   if fc < fv(n)
      vk = vc; fk = fc;
      how = 'contract';
   else
      for j = 2:n
         v(:,j) = (v(:,1) + v(:,j))/2;
         x(:) = v(:,j); fv(j) = eval(evalstr);
      end
cnt = cnt + n-1;
      vk = (v(:,1) + v(:,n+1))/2;
      x(:) = vk; fk = eval(evalstr);
   cnt = cnt + 1;
      how = 'shrink  ';
```

```
   end
end
v(:,n+1) = vk;
fv(n+1) = fk;
[fv,j] = sort(fv);
v = v(:,j);

   if prnt & (cnt-oldcnt)>prntstep
      home
      oldcnt=cnt;
      cnt
      disp(how)
      disp(' ')
      v
      fv
   end
 end
x(:) = v(:,1);
if prnt, format, end
if cnt>=maxit
if prnt >= 0
disp(['Warning: Maximum number of iterations (' ...
      ,num2str(maxit),') has been exceeded']);
disp( '         (increase maxit.')
end
end
```

MaxLa.m

```
     % Matlab function MaxLa.m        updated March 28, 1992
function int =MaxLa(param,y,m,x,A,D,a,offset,dist)
     % MaxLa is called by fminjim which requires the parameters to
     % optimize to appear first in the function call.  This function
     % reorginizes the parameters so that Laplace.m can use them
```

```
     % correctly.  MaxLa also returns the negative of the log-Laplace
     % approximation since fminjim minimizes functions.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
q=min(size(x));
ndisp=sum(a>0);
beta=param(1:q,1);
theta=x*beta+offset;
dispa=param(q+1:q+ndisp,1)'.*(a>0);
if all(dispa>0),
  D=[ones(1,a(1))./dispa(1) ones(1,a(2))./dispa(2)];
  [int w L]=Laplace(y,m,theta,A,D,a,dispa,dist);
elseif dispa(1)>0,
disp('Tau-hat')
dispa(2)
   A1=A(:,1:a(1));
   D=[ones(1,a(1))./dispa(1)];
   [int w L]=Laplace(y,m,theta,A1,D,a(1),dispa(1),dist);
   int=int+1-exp(-dispa(2)*1000);
elseif ndisp>1,
   if dispa(2)>0,
disp('Sig-hat')
dispa(1)
     A2=A(:,a(1)+1:a(1)+a(2));
     D=[ones(1,a(2))./dispa(2)];
     [int w L]=Laplace(y,m,theta,A2,D,a(2),dispa(2),dist);
     int=int+1-exp(-dispa(1)*1000);
    else
disp('Both')
dispa
     int=y'*theta-sum(B1(theta,m,dist));
     int=int+2-exp(-dispa(1)*1000)-exp(-dispa(2)*1000);
    end;
else,
```

```
disp('Sig-hat')
dispa(1)
    int=y'*theta-sum(B1(theta,m,dist))+1-exp(-dispa(1)*1000);
end;
int=-int;
end;
```

MaxLab.m

```
     % Matlab function MaxLab.m       updated March 28, 1992
function [int, w] =MaxLab(param,y,m,theta,A,D,a,dist)
     % MaxLab is the same as MaxLa except that beta has been pulled
     % out of param.  So param holds only the dispersion components.
     % This function is used to optimize the dispersion components
     % for a fixed beta point.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
ndisp=sum(a>0);
dispa=param'.*(a>0);
if all(dispa>0),
  D=[ones(1,a(1))./dispa(1) ones(1,a(2))./dispa(2)];
  [int w L]=Laplace(y,m,theta,A,D,a,dispa,dist);
elseif dispa(1)>0,
disp('Tau-hat')
dispa(2)
   A1=A(:,1:a(1));
   D=[ones(1,a(1))./dispa(1)];
   [int w L]=Laplace(y,m,theta,A1,D,a(1),dispa(1),dist);
   int=int+1-exp(-dispa(2)*1000);
 elseif ndisp>1,
   if dispa(2)>0,
disp('Sig-hat')
dispa(1)
```

```
      A2=A(:,a(1)+1:a(1)+a(2));
      D=[ones(1,a(2))./dispa(2)];
      [int w L]=Laplace(y,m,theta,A2,D,a(2),dispa(2),dist);
      int=int+1-exp(-dispa(1)*1000);
    else
disp('Both')
dispa
      int=y'*theta-sum(B1(theta,m,dist));
      int=int+2-exp(-dispa(1)*1000)-exp(-dispa(2)*1000);
    end;
else,
disp('Sig-hat')
dispa(1)
    int=y'*theta-sum(B1(theta,m,dist))+1-exp(-dispa(1)*1000);
end;
int=-int;
end;
```

MaxGH.m

```
     % function MaxGH.m          updated March 28, 1992
function int=MaxGH(param,y,m,x,A,D,a,offset,dist,GHstuff)
     % MaxGH perfroms the same role as MaxLa for G-H optimization.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
M=GHstuff(1);
wts=GHstuff(2:M+1,:);
nds=GHstuff(M+2:M+M+1,:);
q=min(size(x));
ndisp=sum(a>0);
beta=param(1:q,1);
theta=x*beta;
dispa=param(q+1:q+ndisp,1)'.*(a>0);
```

```
if all(dispa>0),
  D=[ones(1,a(1))./dispa(1) ones(1,a(2))./dispa(2)];
  [int La]=GHint(y,m,theta,A,D,a,dispa,dist,M,wts,nds);
  int=int+La;
elseif dispa(1)>0,
disp('Tau-hat')
dispa(2)
   A1=A(:,1:a(1));
   D=[ones(1,a(1))./dispa(1)];
   [int La]=GHint(y,m,theta,A1,D,a(1),dispa(1),dist,M,wts,nds);
   int=int+La-exp(-dispa(2)*1000);
elseif ndisp>1,
  if dispa(2)>0,
disp('Sig-hat')
dispa(1)
   A2=A(:,a(1)+1:a(1)+a(2));
   D=[ones(1,a(2))./dispa(2)];
   [int La]=GHint(y,m,theta,A2,D,a(2),dispa(2),dist,M,wts,nds);
   int=int+La-exp(-dispa(1)*1000);
  else,
disp('Both')
dispa
   int=y'*theta-sum(B1(theta,m,dist));
   int=int+2-exp(-dispa(1)*1000)-exp(-dispa(2)*1000);
  end;
else,
disp('Sig-hat')
dispa(1)
  int=y'*theta-sum(B1(theta,m,dist));
  int=int+1-exp(-dispa(1)*1000);
end;
int=-int;
end;
```

MaxGHb.m

```
% function MaxGHb.m        updated March 28, 1992
  function int=MaxGHb(param,y,m,theta,A,D,a,dist,GHstuff)
  % MaxGHb performs similar role as MaxLab for G-H integration.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
M=GHstuff(1);
wts=GHstuff(2:M+1,:);
nds=GHstuff(M+2:M+M+1,:);
dispa=param'.*(a>0);
ndisp=sum(a>0);
if all(dispa>0),
  D=[ones(1,a(1))./dispa(1) ones(1,a(2))./dispa(2)];
  [int La]=GHint(y,m,theta,A,D,a,dispa,dist,M,wts,nds);
  int=int+La;
elseif dispa(1)>0,
disp('Tau-hat')
dispa(2)
   A1=A(:,1:a(1));
   D=[ones(1,a(1))./dispa(1)];
   [int La]=GHint(y,m,theta,A1,D,a(1),dispa(1),dist,M,wts,nds);
   int=int+La-exp(-dispa(2)*1000);
elseif ndisp>1,
  if dispa(2)>0,
disp('Sig-hat')
dispa(1)
   A2=A(:,a(1)+1:a(1)+a(2));
   D=[ones(1,a(2))./dispa(2)];
   [int La]=GHint(y,m,theta,A2,D,a(2),dispa(2),dist,M,wts,nds);
   int=int+La-exp(-dispa(1)*1000);
  else,
disp('Both')
dispa
```

```
    int=y'*theta-sum(B1(theta,m,dist));
    int=int+2-exp(-dispa(1)*1000)-exp(-dispa(2)*1000);
  end;
else,
disp('Sig-hat')
dispa(1)
  int=y'*theta-sum(B1(theta,m,dist));
  int=int-exp(-dispa(1)*1000);
end;
int=-int;
end;
```

E. DATA INPUT AND VARIABLE SETUP

celldata.m

```
% Matlab Script file celldata.m
% ENTER DATA VECTORS.
% This script file is used to "input" the data vectors.
% These are:
%    y - the reponse of counts
%    m - the binomial index for each y.  Set m to a vector of
%        ones if Poisson.
%    x - the fixed effects design matrix.
%    u - random effects varaible at top most level.  Like a SAS
%        class variable.
%    v - random effects varaible at second nesting.  Like a SAS
%        class variable.
%    dist - a scaler denoting Poisson (=1) or binomial (=2)
%           counts.
% Computed variables are:
%    N - number of obsrevations.
%    q - number of parameters in the model.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University

% y is response variable.  Counts for both Poisson and binomial.
y=[178 193 217 ...
   109 112 115 ...
    66  75  80 ...
   118 125 137 ...
   123 146 170 ...
   115 130 133 ...
   200 189 173 ...
    88  76  90 ...
```

```
   121 124 136];
y=y';

% create vector of mijk's, called m.  If model says y given w is
% Poisson, then mijk=1. If binomial, then mijk is the binomial index
% for yijk.
m=ones(length(y),1)*400;

% indicates model assumes Poisson (dist=1) or binomial (dist=2).
dist=2;
% create fixed effects design matrix, x.
N=length(y);
x=[ones(N,1)];
q=min(size(x));  % q is the number of beta parameters in the model.

% enter levels for which nesting occurs.
% u and v denote levels of nesting.
u=[1 1 1 2 2 2 3 3 3 4 4 4 5 5 5 6 6 6 ...
   7 7 7 8 8 8 9 9 9];
u=u';
v=[1 2 3 1 2 3 1 2 3 1 2 3 1 2 3 1 2 3 ...
   1 2 3 1 2 3 1 2 3];
v=v';
```

revert.m

```
% Matlab Script file revert.m
% ENTER DATA VECTORS.
% This script file is used to "input" the data vectors.
% These are:
%    y - the reponse of counts
%    m - the binomial index for each y.  Set m to a vector of
%        ones if Poisson.
%    x - the fixed effects design matrix.
```

```
%     u - random effects varaible at top most level.  Like a SAS
%         class variable.
%     v - random effects varaible at second nesting.  Like a SAS
%         class variable.
%     dist - a scaler denoting Poisson (=1) or binomial (=2)
%            counts.
% Computed variables are:
%     N - number of obsrevations.
%     q - number of parameters in the model.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University

% y is response variable.  Counts for both Poisson and binomial.
y=[22 60 98 60 22 23 ...
   23 59 78 82 44 21 ...
   35 54 50 59 33 25 ...
   19 45 26 39 33 10 ...
   17 25 17 44 26 8 ...
   16 24 31 30 23 9 ...
   23 27 28 41 28 16 ...
   22 23 37 37 21 19 ...
   14 21 35 43 30 13];
y=y';

% create vector of mijk's, called m.  If model says y given w is
% Poisson, then mijk=1. If binomial, then mijk is the binomial index
% for yijk.
m=ones(length(y),1);

% indicates model assumes Poisson (dist=1) or binomial (dist=2).
dist=1;
% create fixed effects design matrix, x.
dosei=[0 100 333 1000 3333 10000];
dose=[dosei dosei dosei dosei dosei dosei dosei dosei dosei];
```

```
dose=dose';
d=log(dose+1);
dind=~d;
N=length(y);
xfull=[ones(N,1) d d.*d dind];
xsmall=[ones(N,1)];
x=xsmall;   %for develpoement purposes for now.
q=min(size(x));  % q is the number of beta parameters in the model.

% enter levels for which nesting occurs.
% u and v denote levels of nesting.
u=[1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 ...
   2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 2 ...
   3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3 3];
u=u';
%v=[1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 ...
%1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 ...
%1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 ];
v=[1 1 1 1 1 1 2 2 2 2 2 2 3 3 3 3 3 3 ...
   1 1 1 1 1 1 2 2 2 2 2 2 3 3 3 3 3 3 ...
   1 1 1 1 1 1 2 2 2 2 2 2 3 3 3 3 3 3 ];
v=v';
```

salamander1.m

```
% Matlab Script file salamander1.m
% ENTER DATA VECTORS.
% This script file is used to "input" the data vectors.
% These are:
%    y - the reponse of counts
%    m - the binomial index for each y.  Set m to a vector of
%        ones if Poisson.
%    x - the fixed effects design matrix.
%    u - random effects varaible at top most level.  Like a SAS
```

```
%         class variable.
%     v - random effects varaible at second nesting.  Like a SAS
%         class variable.
%     dist - a scaler denoting Poisson (=1) or binomial (=2)
%            counts.
% Computed variables are:
%     N - number of obsrevations.
%     q - number of parameters in the model.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University

% y is response variable.  Counts for both Poisson and binomial.
y=[1 1 1 1 1 1 1 0 1 1 1 1 1 1 1 0 1 1 0 1 1 1 1 1 1 1 1 1 1 1 ...
   1 0 0 0 0 1 0 1 0 0 1 0 0 1 1 0 1 0 1 0 1 0 1 1 1 1 0 1 1 0 ...
   0 0 0 0 0 1 0 0 1 1 1 0 0 1 0 1 1 0 1 0 0 0 0 0 0 1 0 1 1 0 ...
   0 1 1 1 1 0 1 0 1 0 1 1 1 1 0 0 0 0 1 1 0 1 1 1 1 0 1 0 0 0 ...
   1 1 0 1 1 0 1 1 0 0 1 1 1 0 1 0 0 1 0 0 1 0 1 0 0 0 1 1 0 0 ...
   1 1 1 1 0 1 1 0 1 0 1 0 0 1 1 0 1 0 1 1 0 0 1 0 1 0 1 0 1 0 ...
   0 0 0 0 0 1 1 1 0 0 0 0 0 1 0 0 0 1 1 0 0 0 0 0 0 0 1 1 1 1 0 ...
   1 1 0 1 1 0 0 0 1 0 1 1 0 1 1 1 0 0 1 1 1 0 0 1 1 1 0 0 1 0 ...
   1 0 1 1 0 1 0 1 0 1 1 0 1 1 1 0 1 0 1 0 1 0 1 0 1 1 0 1 0 0 ...
   0 0 1 0 0 1 1 1 0 1 1 0 1 1 1 1 0 1 0 1 1 0 1 1 1 1 0 0 1 1 ...
   0 0 1 1 0 0 1 0 1 1 0 0 1 1 1 1 1 1 0 0 1 0 0 0 0 0 1 1 1 0 1 ...
   0 1 0 1 0 0 0 0 0 0 1 1 1 0 1 0 0 0 0 0 0 0 1 1 1 0 0 0 0 0];
y=y';

% create vector of mijk's, called m.  If model says y given w is
% Poisson, then mijk=1. If binomial, then mijk is the binomial index
% for yijk.
m=ones(length(y),1);

% indicates model assumes Poisson (dist=1) or binomial (dist=2).
dist=2;
% create fixed effects design matrix, x.
```

```
N=length(y);
X1=ones(length(y),1);
X2=[ones(1,120) 2*ones(1,120) 3*ones(1,120)]';
X3=[zeros(1,120) ones(1,240)]';
X4=[zeros(1,60) ones(1,60) zeros(1,60) ones(1,60) zeros(1,60) ...
    ones(1,60)]';
X5=[zeros(1,5) ones(1,5) zeros(1,5) ones(1,5) zeros(1,5) ...
    ones(1,5) ones(1,5) zeros(1,5) ones(1,5) zeros(1,5) ...
    ones(1,5) zeros(1,5)];
X5=[X5 X5 X5 X5 X5 X5]';
X6=X4.*X5;
XA=[X1 X4 X5 X6];
XB=[X1 X4 X5 X6 X3];
x=XA;
q=min(size(x));  % q is the number of beta parameters in the model.

% enter levels for which nesting occurs.
% u and v denote levels of nesting.
u=[1:5 1:5 1:5 1:5 1:5 1:5 6:10 6:10 6:10 6:10 6:10 6:10 ...
   11:15 11:15 11:15 11:15 11:15 11:15 16:20 16:20 16:20 ...
   16:20 16:20 16:20 21:25 21:25 21:25 21:25 21:25 21:25 ...
   26:30 26:30 26:30 26:30 26:30 26:30 31:35 31:35 31:35 ...
   31:35 31:35 31:35 36:40 36:40 36:40 36:40 36:40 36:40 ...
   41:45 41:45 41:45 41:45 41:45 41:45 46:50 46:50 46:50 ...
   46:50 46:50 46:50 51:55 51:55 51:55 51:55 51:55 51:55 ...
   56:60 56:60 56:60 56:60 56:60 56:60]';
v=[1 5 2 4 3 14 15 11 12 13 5 3 1 2 4 11 13 14 15 12 4 1 3 5 2 ...
   15 12 13 14 11 19 18 16 20 17 9 8 6 7 10 20 19 17 18 16 7 9 ...
   10 6 8 16 17 20 19 18 8 6 9 10 7 9 7 8 10 6 19 16 17 20 18 7 ...
   9 6 8 10 20 17 19 18 16 10 6 7 9 8 18 20 16 19 17 15 14 11 13 ...
   12 2 1 4 3 5 13 15 12 11 14 4 2 5 1 3 12 11 15 14 13 1 5 3 4 2];
v=[v 20+v 40+v]';
y=y(1:120);
x=x(1:120,:);
u=u(1:120,:);
```

```
v=v(1:120,:);
m=m(1:120,:);
N=length(y);
q=min(size(x));
```

setup.m

```
    % Matlab script file: setup.m.
    % Takes the random effects vectors, u and v, and creates their
    % corresponding incidence matrices A1 and A2, as in thesis.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University

% A1 and vm are incidence matrices for u and v, as in proposal.
A1=zeros(N,max(u));
vm=zeros(N,max(v));
for i=1:N
  A1(i,u(i))=1;
  vm(i,v(i))=1;
end

% A2 is crossed incidence matrix for A1 and vm, used for
%  getting marginal totals.
A2=[];
for i=1:min(size(A1))
  for j=1:min(size(vm))
   A2=[A2 A1(:,i).*vm(:,j)];
  end
end
clear vm;
a=[min(size(A1)) min(size(A2))];
```

setup2.m

```
% Matlab script file: setup2.m.
% Takes the random effects vectors, u and v, and creates their
% corresponding incidence matrices A1 and A2, as in thesis.
% Used for crossed random effects aspect of salamander mating
% experiments.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University

% A1 and vm are incidence matrices for u and v, as in proposal.
A1=zeros(N,max(u));
A2=zeros(N,max(v));
for i=1:N
  A1(i,u(i))=1;
  A2(i,v(i))=1;
end
a=[min(size(A1)) min(size(A2))];
```

F. OPTIMIZATION ROUTINES

optIGLS.m

```
% Matlab script file  optIGLS.m   updated Sept 8,1992
% Script file to obtain IGLS estimates for fixed effects and
% dispersion components, using methods described in McCullagh
% and Nelder (1989, Chapter 14).  Program assumes script file
% initial has been run.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
begin=clock;
[betagls dispgls]=gls(y,m,x,betahi,A1,A2,a,dist);
[mu Vmu]=Bw(x*betagls,m,dist);
W=diag(Vmu);
D=diag([ones(1,a(1))*dispgls(1) ones(1,a(2))*dispgls(2)]);
A=[A1 A2];
glscov=inv(x'*inv(inv(W)+A*D*A')*x);
timegls=etime(clock,begin)
```

optREML.m

```
% Matlab script file optREML.m    updated Sept 8, 1992
% This script file obtaains REML estimates of the fixed effects,
% dispersion components, and the "mode" using formulas in Schall
% (1991).  This program assumes optIGLS has been run for
% starting values.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
begin=clock;
[betar dispr moder]= ...
   reml(y,m,x,betagls,A,a,dispgls,zeros(sum(a),1),dist,25);
```

```
[mu Vmu]=Bw(x*betar+A*moder,m,dist);
W=diag(Vmu);
D=diag([ones(1,a(1))*dispr(1) ones(1,a(2))*dispr(2)]);
covreml=inv(x'*inv(inv(W)+A*D*A')*x);
timereml=etime(clock,begin)
```

optLa.m

```
    % Matlab script file optLa.m      updated Sept 8, 1992
    % Script file to find Laplace approximate MLE estimates of fixed
    % effects and dispersion components.  Returns "mode" at the MLE
    % solution.  Program assume optIGLS has been run for starting
    % values.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
begin=clock;
D=[ones(1,a(1))./dispgls(1) ones(1,a(2))./dispgls(2)];
[Lapar its]=fminjim('MaxLa',[betagls' dispgls']',20, ...
   1.e-3,.0001,y,m,x,[A1 A2],D,a,zeros(sum(a),1),dist);
betaLa=Lapar(1:q,1);
dispLa=Lapar(q+1:q+2,1);
if dispLa(1)==-1,
  DA2=ons(1,a(2))./dispLa(2);
  [Lapar its]=fminjim('MaxLa',[betaLa' dispLa(2)]',20, ...
     1.e-3,.0001,y,m,x,A2,DA2,[0 a(2)],zeros(sum(a),1),dist);
  betaLa=Lapar(1:q,1);
  dispLa=[0 Lapar(q+1,1)];
  Lapar=[betaLa' dispLa]';
elseif dispLa(2)==-1,
  DA1=ons(1,a(1))./dispLa(1);
  [Lapar its]=fminjim('MaxLa',[betaLa' dispLa(1)]',20, ...
     1.e-3,.0001,y,m,x, A1,DA1,[a(1) 0],zeros(sum(a),1),dist);
  betaLa=Lapar(1:q,1);
```

```
  dispLa=[Lapar(q+1,1) 0];
  Lapar=[betaLa' dispLa]';
end;
timeLa=etime(clock,begin)
```

optGH.m

```
    % Matlab script file optGH.m     updated Nov 9, 1992
    % Script file to find Num. Int. approximate MLE estimates of
    % fixed effects and dispersion components.  Returns "mode" at
    % the MLE solution.  Program assume optIGLS has been run (for
    % starting values) and uses GHstuff.
%   James Pratt, 1-10-91
%   Dept. of Statistics
%   Oregon State University
begin=clock;
D=[ones(1,a(1))./dispgls(1) ones(1,a(2))./dispgls(2)];
[GHpar its]=fminjim('MaxGH',[betagls' dispgls']',20, ...
   1.e-3,.0001,y,m,x,[A1 A2],D,a,zeros(sum(a),1),dist,GHstuff);
betaGH=GHpar(1:q,1);
dispGH=GHpar(q+1:q+2,1);
if dispGH(1)==-1,
  DA2=ons(1,a(2))./dispGH(2);
  [GHpar its]=fminjim('MaxGH',[betaGH' dispGH(2)]',20, ...
     1.e-3,.0001,y,m,x,A2,DA2,[0 a(2)],zeros(sum(a),1),dist,GHstuff);
  betaGH=GHpar(1:q,1);
  dispGH=[0 GHpar(q+1,1)];
  GHpar=[betaGH' dispGH]';
elseif dispGH(2)==-1,
  DA1=ons(1,a(1))./dispGH(1);
  [GHpar its]=fminjim('MaxGH',[betaGH' dispGH(1)]',20, ...
     1.e-3,.0001,y,m,x,A1,DA1,[a(1) 0],zeros(sum(a),1),dist,GHstuff);
  betaGH=GHpar(1:q,1);
  dispGH=[GHpar(q+1,1) 0];
```

```
  GHpar=[betaGH' dispGH]';
end;
timeLa=etime(clock,begin)
```

opt.m

```
    % Matlab script file opt.m    updated March 28, 1992
initial
qlcov=covbetahi;
[betagls dispgls]=gls(y,m,x,betahi,A1,A2,a,dist);
betagls
dispgls
glscov=inv(x'*inv(inv(W)+(dispgls(1)*A1*A1'+dispgls(2)*A2*A2'))*x);
%disppar=dispgls;   % Comment these lines out if running simulations;
%beta=betahi;       % Put the true values in the variables;
D=[ones(1,a(1))./disppar(1) ones(1,a(2))./disppar(2)];
mode=zeros(sum(a),1);
offset=zeros(y);

% Optimize Laplace approx. over full parameter sapce;
[Lapar its]=fminjim('MaxLa',[beta' disppar']',20,1.e-3, ...
   .0001,y,m,x,[A1 A2],D,a,offset,dist);
betaLa=Lapar(1:q,1);
dispLa=Lapar(q+1:q+2,1)';
if dispLa(1)==-1,
   DA2=ones(1,a(2))./dispLa(2);
   [Lapar its]=fminjim('MaxLa',[betaLa' dispLa(2)]',20, ...
      1.e-3,.001,y,m,x,A2,DA2,[0 a(2)],offset,dist);
   betaLa=Lapar(1:q,1);
   dispLa=[0 Lapar(q+1,1)];
   Lapar=[betaLa' dispLa]';
elseif dispLa(2)==-1,
   DA1=ones(1,a(1))./dispLa(1);
   [Lapar its]=fminjim('MaxLa',[betaLa' dispLa(1)]',20, ...
```

```
      1.e-3,.001,y,m,x,A1,DA1,[a(1) 0],offset,dist);
   betaLa=Lapar(1:q,1);
   dispLa=[Lapar(q+1,1) 0];
   Lapar=[betaLa' dispLa]';
end;
if min(dispLa)==-1,
   betaLa=betahi;
   dispLa=[0 0];
   Lapar=[betaLa' dispLa]';
end;
LaMax =MaxLa(Lapar,y,m,x,[A1 A2],D,a,offset,dist);

% Optimize the dispersion components at the fixed beta point;
[dispbeta its]=fminjim('MaxLab',disppar,20,1.e-3,.0001, ...
    y,m,x*beta,[A1 A2],D,a,dist);

if dispbeta(1)==-1,
   DA2=ones(1,a(2))./dispbeta(2);
   [dispbeta its]=fminjim('MaxLab',dispbeta(2),20, ...
      1.e-3,.001,y,m,x*beta,A2,DA2,[0 a(2)],dist);
   dispbeta=[0 dispbeta]';
elseif dispbeta(2)==-1,
   DA1=ones(1,a(1))./dispbeta(1);
   [dispbeta its]=fminjim('MaxLab',dispbeta(1),20, ...
      1.e-3,.001,y,m,x*beta,A1,DA1,[a(1) 0],dist);
     dispbeta=[dispbeta 0]';
  end;
if min(dispbeta)==-1,
   dispbeta=[0 0];
end;
Labeta =MaxLa([beta' dispbeta']',y,m,x,[A1 A2],D,a,offset,dist);

GHwts;
% Optimize Likelihood using 2 node quad and whole parameter space;
GHstuff=[2 w2' n2']';
```

```
[GHpar its]=fminjim('MaxGH',[beta' disppar']',20, ...
   1.e-3,.0001,y,m,x,[A1 A2],D,a,offset,dist,GHstuff);
betaGH2=GHpar(1:q,1);
dispGH2=GHpar(q+1:q+2,1)';

if dispGH2(1)==-1,
   DA2=ones(1,a(2))./dispGH2(2);
   [GHpar its]=fminjim('MaxGH',[betaGH2' dispGH2(2)]',20, ...
      1.e-3,.001,y,m,x,A2,DA2,[0 a(2)],offset,dist,GHstuff);
   betaGH2=GHpar(1:q,1);
   dispGH2=[0 GHpar(q+1,1)];
   GHpar=[betaGH2' dispGH2]';
elseif dispGH2(2)==-1,
   DA1=ones(1,a(1))./dispGH2(1);
   [GHpar its]=fminjim('MaxGH',[betaGH2' dispGH2(1)]',20, ...
      1.e-3,.001,y,m,x,A1,DA1,[a(1) 0],offset,dist,GHstuff);
   betaGH2=GHpar(1:q,1);
   dispGH2=[GHpar(q+1,1) 0];
   GHpar=[betaGH2' dispGH2]';
end;
if min(dispGH2)==-1,
   betaGH2=betahi;
   dispGH2=[0 0];
   GHpar=[betaGH2' dispGH2]';
end;
GH2Max =MaxGH(GHpar,y,m,x,[A1 A2],D,a,offset,dist,GHstuff);

[dispGH2b its]=fminjim('MaxGHb',disppar,20,1.e-3,.0001, ...
    y,m,x*beta,[A1 A2],D,a,dist,GHstuff);

if dispGH2b(1)==-1,
   DA2=ones(1,a(2))./dispGH2b(2);
   [dispGH2b its]=fminjim('MaxGHb',dispGH2b(2),20, ...
       1.e-3,.001,y,m,x*beta,A2,DA2,[0 a(2)],dist,GHstuff);
   dispGH2b=[0 dispGH2b]';
```

```
elseif dispGH2b(2)==-1,
   DA1=ones(1,a(1))./dispGH2b(1);
   [dispGH2b its]=fminjim('MaxGHb',dispGH2b(1),20, ...
       1.e-3,.001,y,m,x*beta,A1,DA1,[a(1) 0],dist,GHstuff);
     dispGH2b=[dispGH2b 0]';
  end;
if min(dispGH2b)==-1,
   dispGH2b=[0 0]';
end;
GH2beta=MaxGH([beta' dispGH2b']',y,m,x,[A1 A2],D,a,offset, ...
          dist,GHstuff);

% Optimize using 5 node quad;
GHstuff=[5 w5' n5']';
[GHpar its]=fminjim('MaxGH',[beta' disppar']',20, ...
    1.e-3,.0001,y,m,x,[A1 A2],D,a,offset,dist,GHstuff);
betaGH5=GHpar(1:q,1);
dispGH5=GHpar(q+1:q+2,1)';

if dispGH5(1)==-1,
   DA2=ones(1,a(2))./dispGH5(2);
   [GHpar its]=fminjim('MaxGH',[betaGH5' dispGH5(2)]',20, ...
      1.e-3,.001,y,m,x,A2,DA2,[0 a(2)],offset,dist,GHstuff);
   betaGH5=GHpar(1:q,1);
   dispGH5=[0 GHpar(q+1,1)];
   GHpar=[betaGH5' dispGH5]';
elseif dispGH5(2)==-1,
   DA1=ones(1,a(1))./dispGH5(1);
   [GHpar its]=fminjim('MaxGH',[betaGH5' dispGH5(1)]',20, ...
      1.e-3,.001,y,m,x,A1,DA1,[a(1) 0],offset,dist,GHstuff);
   betaGH5=GHpar(1:q,1);
   dispGH5=[GHpar(q+1,1) 0];
   GHpar=[betaGH5' dispGH5]';
end;
if min(dispGH5)==-1,
```

```
  betaGH5=betahi;
  dispGH5=[0 0];
  GHpar=[betaGH5' dispGH5]';
end;

GH5Max =MaxGH(GHpar,y,m,x,[A1 A2],D,a,offset,dist,GHstuff);
[dispGH5b its]=fminjim('MaxGHb',disppar,20,1.e-3,.0001, ...
    y,m,x*beta,[A1 A2],D,a,dist,GHstuff);

if dispGH5b(1)==-1,
   DA2=ones(1,a(2))./dispGH5b(2);
   [dispGH5b its]=fminjim('MaxGHb',dispGH5b(2),20, ...
      1.e-3,.001,y,m,x*beta,A2,DA2,[0 a(2)],dist,GHstuff);
   dispGH5b=[0 dispGH5b]';
elseif dispGH5b(2)==-1,
   DA1=ones(1,a(1))./dispGH5b(1);
   [dispGH5b its]=fminjim('MaxGHb',dispGH5b(1),20, ...
      1.e-3,.001,y,m,x*beta,A1,DA1,[a(1) 0],dist,GHstuff);
     dispGH5b=[dispGH5b 0]';
  end;
if min(dispGH5b)==-1,
   dispGH5b=[0 0]';
end;

GH5beta =MaxGH([beta' dispGH5b']',y,m,x,[A1 A2],D,a,offset, ...
              dist,GHstuff);
```

opt2.m

```
    % Matlab script file opt2.m    updated Feb 24, 1992
indata
setup
str1=38991;
str2=46901;
```

```
str3=10456;
tau1=.25;
tau2=.1;
disppar=[.25 .1]';
if dist==1,
  beta=3;
else,
  beta=1;
end;
numsam=200;
glim=zeros(numsam,3);
ql=zeros(numsam,2);
gls=zeros(numsam,4);
Las=zeros(numsam,9);
GH2=zeros(numsam,7);
GH5=zeros(numsam,7);
GHwts
for i=1:numsam,
  samples
  y=Y;
  opt
disp('opt done')
   lbetahi=y'*x*betahi-sum(B1(x*betahi,m,dist));
   lbeta=y'*x*beta-sum(B1(x*beta,m,dist));
   glim(i,:)=[betahi lbetahi lbeta];
   [mu V]=Bw(x*betahi,m,dist);
   W=diag(V);
   ql(i,:)=[oversig (betahi-beta)/sqrt(oversig*inv(x'*W*x))];
   [mu V]=Bw(x*betagls,m,dist);
   D=dispgls(1)*A1*A1'+dispgls(2)*A2*A2';
   Wgls=inv(inv(diag(V))+D);
   gls(i,:)=[betagls dispgls' (betagls-beta)/sqrt(inv(x'*Wgls*x))];
   Las(i,:)=[betaLa dispLa dispbeta' LaMax Labeta GHLMax GHLbeta];
   GH2(i,:)=[betaGH2 dispGH2 dispGH2b'GH2Max GH2beta];
   GH5(i,:)=[betaGH5 dispGH5 dispGH5b'GH5Max GH5beta];
```

```
   if rem(i,10)==0 & i>9,
      save hists2b beta tau1 tau2 glim ql gls Las GH2 GH5;
   end;
end;
endtime=clock;

save hists2b endtime beta tau1 tau2 glim ql gls Las GH2 GH5;
quit;
```

optimize.m

```
% Matlab script file optimize.m
[betagls1 dispgls1]=gls(y,m,x,betahi,A1,A2,a,dist);
pargls1=[betagls1' dispgls1']';

dispGH=dispgls1+.1;
D=[ones(1,a(1))./dispGH(1) ones(1,a(2))./dispGH(2)];
mode=zeros(sum(a),1);

[Lagls1 modegls1]=MaxLa(pargls1,y,m,x,[A1 A2],D,a,mode,dist);
[Lapar its]=fminjim('MaxLa',[betahi' dispGH']',20,1.e-3, ...
   .01,y,m,x,[A1 A2],D,a,mode,dist);

[LaMax modeMax]=MaxLa(Lapar,y,m,x,[A1 A2],D,a,mode,dist);

betaLa=Lapar(1:q,1);
dispLa=Lapar(q+1:q+2,1)';
GHwts;

[GH2 La mode]=GHint(y,m,x*betaLa,[A1 A2],D,a,dispLa,modeMax, ...
              dist,2,w2,n2);
GHstuff=[2 w2' n2']';
[GHpar its]=fminjim('MaxGH',Lapar,20,1.e-3,.01,y,m,x,[A1 A2], ...
             D,a,modeMax,dist,GHstuff);
```

```
[GHMax modeGH]=MaxGH(GHpar,y,m,x,[A1 A2],D,a,modeMax,dist,GHstuff);

disp('IGLS estimates')
pargls1'
disp('Laplace estimates')
Lapar'
disp('GH estimates')
GHpar'
disp('Log. Laplace est. at IGLS and Laplace, with GH correction')
[-Lagls1 -LaMax GH2 -GHMax]
```

optmore.m

```
    % Matlab script file optmore.m
initial
[betagls dispgls]=gls1(y,m,x,betahi,A1,A2,a,dist);
pargls=[betagls' dispgls']';
[Lagls modegls]=MaxLa(pargls,y,m,x,[A1 A2],D,a,mode,dist);

dispGH=dispgls+.1;
D=[ones(1,a(1))./dispGH(1) ones(1,a(2))./dispGH(2)];
D=[ones(1,a(1))./disppar(1) ones(1,a(2))./disppar(2)];
mode=zeros(sum(a),1);

[Lapar its]=fminjim('MaxLa',[beta' disppar']',20, ...
              1.e-3,.0001,y,m,x,[A1 A2],D,a,mode,dist);

betaLa=Lapar(1:q,1);
dispLa=Lapar(q+1:q+2,1)';
if dispLa(1)==-1,
   DA2=ones(1,a(2))./dispLa(2);
   [Lapar its]=fminjim('MaxLa',[betaLa' dispLa(2)]',20, ...
                 1.e-3,.001,y,m,x,A2,DA2,[0 a(2)],mode,dist);
   betaLa=Lapar(1:q,1);
```

```
   dispLa=[0 Lapar(q+1,1)];
   Lapar=[betaLa' dispLa]';
elseif dispLa(2)==-1,
   DA1=ones(1,a(1))./dispLa(1);
   [Lapar its]=fminjim('MaxLa',[betaLa' dispLa(1)]',20, ...
                1.e-3,.001,y,m,x,A1,DA1,[a(1) 0],mode,dist);
   betaLa=Lapar(1:q,1);
   dispLa=[Lapar(q+1,1) 0];
   Lapar=[betaLa' dispLa]';
end;

[LaMax modeMax]=MaxLa(Lapar,y,m,x,[A1 A2],D,a,mode,dist);
[GHLMax La mode]=GHint(y,m,x*betaLa,[A1
A2],D,a,dispLa,modeMax,dist,3,w3,n3);

[dispbeta its]=fminjim('MaxLab',disppar,20,1.e-3,.0001, ...
                y,m,x*beta,[A1 A2],D,a,modeMax,dist);
if dispbeta(1)==-1,
   DA2=ones(1,a(2))./dispbeta(2);
   [dispbeta its]=fminjim('MaxLab',dispbeta(2),20, ...
                  1.e-3,.001,y,m,x*beta,A2,DA2,[0 a(2)],mode,dist);
   dispbeta=[0 dispbeta]';
elseif dispbeta(2)==-1,
   DA1=ones(1,a(1))./dispbeta(1);
   [dispbeta its]=fminjim('MaxLab',dispbeta(1),20, ...
                   1e-3,.001,y,m,x*beta,A1,DA1,[a(1) 0],mode,dist);
     dispbeta=[dispbeta 0]';
  end;

[Labeta modebeta]=MaxLa([beta' dispbeta']',y,m,x,[A1 A2],D,a, ...
                    modeMax,dist);
[GHLbeta Lab mode]=GHint(y,m,x*beta,[A1 A2],D,a,dispbeta', ...
                     modebeta,dist,3,w3,n3);
```

G. ROUTINES FOR PROFILE LIKELIHOODS

profLA.m

```
% script file profLA.m              updated Sep 9, 1992
% For a given beta estimate betaLa, finds profile likelihood of
% Laplace by maximizing the objective functions along a grid of
% beta values. This script file assumes one has created x0 and
% beta0, the parameter and column from design matrix for which
% profile likelihood is to be obtained, and x1 and beta1 ,
% nuisance parameters and rest of the design matrix.  If the
% beta parameter space is scaler, then x1=[].
% beta0 should contain the point that the grid will be symmetric
% around.
midindex=6;
lenprof=2*midindex-1;
betagrid=zeros(lenprof,1);
Lagrid=zeros(lenprof,1);
step=se0*3/(midindex-1);
ind=[[midindex:-1:1] [(midindex+1):lenprof]];
for i=1:lenprof,
  index=ind(i);
  betagrid(index)=beta0-(midindex-index)*step;
  dispbeta=dispLa;
  beta=betagrid(index);
  if min(size(x1))==0,
    offset=zeros(y);
    [dispbeta its]=fminjim('MaxLab',dispbeta,20, ...
                    1.e-1,.001,y,m,x0*beta,[A1 A2],D,a,dist);
    if dispbeta(1)==-1,
      Da2=ones(1,a(2))./dispbeta(2);
      [dispbeta its]=fminjim('MaxLab',dispbeta(2),20, ...
                      1.e-1,.001,y,m,x0*beta,A2,Da2,[0 a(2)],dist);
      dispbeta=[0 dispbeta]';
```

```
  elseif dispbeta(2)==-1,
    Da1=ones(1,a(1))./dispbeta(1);
    [dispbeta its]=fminjim('MaxLab',dispbeta(1),20, ...
                    1.e-1,.001,y,m,x0*beta,A1,Da1,[a(1) 0],dist);
    dispbeta=[dispbeta 0]';
  end;
  if min(dispbeta)==-1,
    dispbeta=[0 0]';
  end;
  Labeta =MaxLa([beta' dispbeta']',y,m,x,[A1 A2],D,a,offset,dist);
  Lagrid(index)=Labeta;
else,
  offset=x0*beta;
  [LAbeta its]=fminjim('MaxLa',[beta1' dispbeta']',20, ...
                1.e-1,.001,y,m,x1,[A1 A2],D,a,offset,dist);
  betabeta=LAbeta(1:q-1,1);
  dispbeta=LAbeta(q:q+1,1);
  if dispbeta(1)==-1,
    Da2=ones(1,a(2))./dispbeta(2);
    [LAbeta its]=fminjim('MaxLa',[betabeta' dispbeta(2)]',20, ...
                  1.e-1,.001,y,m,x1,A2,Da2,[0 a(2)],offset,dist);
    betabeta=LAbeta(1:q-1,1);
    dispbeta=[0 LAbeta(q,1)]';
    LAbeta=[betabeta' dispbeta']';
  elseif dispbeta(2)==-1,
    Da1=ones(1,a(1))./dispbeta(1);
    [LAbeta its]=fminjim('MaxLa',[betabeta' dispbeta(1)]',20, ...
                  1.e-1,.001,y,m,x1,A1,Da1,[a(1) 0],offset,dist);
    betabeta=LAbeta(1:q-1,1);
    dispbeta=[LAbeta(q,1) 0]';
    LAbeta=[betabeta' dispbeta']';
  end;
  if min(dispbeta)==-1,
    dispbeta=[0 0]';
  end;
```

```
    Labeta =MaxLa([betabeta' dispbeta']',y,m,x1,[A1 A2],D,a, ...
                offset,dist);
    Lagrid(index)=Labeta;
  end;
end;
LRLA=2*(min(Lagrid)-Lagrid);
save profile
```

profGH.m

```
    % script file profGH.m    Updated June 10, 1992
    % For a given beta estimate betaLa, finds profile likelihood of
    % GH by maximizing the objective functions along a grid of beta
    % values.  This assumes that beta is scaler.  This file strictly
    % for revertant colony profile for gamma and obtains 9-node GH.
load revert2
GH9grid=zeros(31,1);
GHstuff=[9 w9' n9']';
for index=1:31,
  index
  dispbeta=dispLa';
  beta=betagrid(index);
reset
  if min(size(x1))==0,
    offset=zeros(y);
  [dispbeta its]=fminjim('MaxGHb',dispbeta,20,1.e-3,.0001, ...
                  y,m,x*beta,[A1 A2],D,a,dist,GHstuff);
  if dispbeta(1)==-1,
    Da2=ones(1,a(2))./dispbeta(2);
    [dispbeta its]=fminjim('MaxGHb',dispbeta(2),20, ...
                1.e-3,.0001,y,m,x*beta,A2,Da2,[0 a(2)],dist,GHstuff);
    dispbeta=[0 dispbeta]';
  elseif dispbeta(2)==-1,
    Da1=ones(1,a(1))./dispbeta(1);
```

```
    [dispbeta its]=fminjim('MaxGHb',dispbeta(1),20, ...
                1.e-3,.0001,y,m,x*beta,A1,Da1,[a(1) 0],dist,GHstuff);
    dispbeta=[dispbeta 0]';
  end;
  if min(dispbeta)==-1,
    dispbeta=[0 0]';
  end;
  GH9beta =MaxGH([beta' dispbeta']',y,m,x,[A1 A2],D,a,offset, ...
            dist,GHstuff);
  GH9grid(index)=GH9beta;
else,
offset=x0*beta;
  [GHbeta its]=fminjim('MaxGH',[beta1'dispbeta']',20, ...
                1.e-3,.0001,y,m,x1,[A1 A2],D,a,offset,dist,GHstuff);
  betabeta=GHbeta(1:q-1,1);
  dispbeta=GHbeta(q:q+1,1);
  if dispbeta(1)==-1,
    Da2=ones(1,a(2))./dispbeta(2);
  [GHbeta its]=fminjim('MaxGH',[betabeta'dispbeta(2)]',20, ...
            1.e-3,.0001,y,m,x1,A2,Da2,[0 a(2)],offset,dist,GHstuff);
  betabeta=GHbeta(1:q-1,1);
  dispbeta=[0 GHbeta(q,1)]';
  GHbeta=[betabeta' dispbeta']';
  elseif dispbeta(2)==-1,
    Da1=ones(1,a(1))./dispbeta(1);
  [GHbeta its]=fminjim('MaxGH',[betabeta'dispbeta(1)]',20, ...
            1.e-3,.0001,y,m,x1,A1,Da1,[a(1) 0],offset,dist,GHstuff);
  betabeta=GHbeta(1:q-1,1);
  dispbeta=[GHbeta(q,1) 0]';
  GHbeta=[betabeta' dispbeta']';
end;
if min(dispbeta)==-1,
   dispbeta=[0 0]';
end;
  GH9beta =MaxGH([betabeta' dispbeta']',y,m,x1,[A1 A2],D,a, ...
```

```
          offset,dist,GHstuff);
  GH9grid(index)=GH9beta;
end;
LRGH9=2*(min(GH9grid)-GH9grid);
save profGH9
end;
quit
```

profile.m

```
     % script file profile.m            updated March 28, 1992
     % For a given beta estimate betaLa, finds profile likelihood of
     % Laplace and GH2 (and IWLS) by maximizing the objective
     % functions along a grid of beta values. This script file
     % assumes one has created x0 and beta0, the parameter and column
     % from design matrix for which profile likelihood is to be
     % obtained, and x1 and beta1 , nuisance parameters and rest of
     % the design matrix.  If the beta parameter space is scaler,
     % then x1=[].  beta0 should contain the point that the grid will
     % be symmetric around.
betagrid=zeros(31,1);
liwls=zeros(31,1);
Lagrid=zeros(31,1);
GH2grid=zeros(31,1);
step=sqrt(glscov0)/5;
index=1;
while index<16,
  betagrid(index,1)=beta0-(16-index)*step;
  betagrid(32-index,1)=beta0+(16-index)*step;
  index=index+1;
end;
betagrid(16)=beta0;
GHstuff=[2 w2' n2']';
for index=1:31,
```

```
  dispbeta=dispLa';
  beta=betagrid(index);
reset
  if min(size(x1))==0,
    offset=zeros(y);
    [dispbeta its]=fminjim('MaxLab',dispbeta,20, ...
                    1.e-3,.0001,y,m,x*beta,[A1 A2],D,a,dist);
    if dispbeta(1)==-1,
      Da2=ones(1,a(2))./dispbeta(2);
      [dispbeta its]=fminjim('MaxLab',dispbeta(2),20, ...
                      1.e-3,.0001,y,m,x*beta,A2,Da2,[0 a(2)],dist);
      dispbeta=[0 dispbeta]';
    elseif dispbeta(2)==-1,
      Da1=ones(1,a(1))./dispbeta(1);
      [dispbeta its]=fminjim('MaxLab',dispbeta(1),20, ...
                      1.e-3,.0001,y,m,x*beta,A1,Da1,[a(1) 0],dist);
      dispbeta=[dispbeta 0]';
    end;
    if min(dispbeta)==-1,
      dispbeta=[0 0]';
    end;
    Labeta =MaxLa([beta' dispbeta']',y,m,x,[A1 A2],D,a,offset,dist);
    Lagrid(index)=Labeta;
    [dispbeta its]=fminjim('MaxGHb',dispbeta,20, ...
                  1.e-3,.0001,y,m,x*beta,[A1 A2],D,a,dist,GHstuff);
    if dispbeta(1)==-1,
      Da2=ones(1,a(2))./dispbeta(2);
      [dispbeta its]=fminjim('MaxGHb',dispbeta(2),20, ...
                1.e-3,.0001,y,m,x*beta,A2,Da2,[0 a(2)],dist,GHstuff);
      dispbeta=[0 dispbeta]';
    elseif dispbeta(2)==-1,
      Da1=ones(1,a(1))./dispbeta(1);
      [dispbeta its]=fminjim('MaxGHb',dispbeta(1),20, ...
                1.e-3,.0001,y,m,x*beta,A1,Da1,[a(1) 0],dist,GHstuff);
      dispbeta=[dispbeta 0]';
```

```
  end;
  if min(dispbeta)==-1,
    dispbeta=[0 0]';
  end;
  GH2beta =MaxGH([beta' dispbeta']',y,m,x,[A1 A2],D,a, ...
              offset,dist,GHstuff);
  GH2grid(index)=GH2beta;
  liwls(index)=y'*x*beta-sum(B1(x*beta,m,dist));
else,
  offset=x0*beta;
  [LAbeta its]=fminjim('MaxLa',[beta1' dispbeta']',20, ...
                1.e-3,.0001,y,m,x1,[A1 A2],D,a,offset,dist);
  betabeta=LAbeta(1:q-1,1);
  dispbeta=LAbeta(q:q+1,1);
  if dispbeta(1)==-1,
    Da2=ones(1,a(2))./dispbeta(2);
    [LAbeta its]=fminjim('MaxLa',[betabeta' dispbeta(2)]',20, ...
                  1.e-3,.0001,y,m,x1,A2,Da2,[0 a(2)],offset,dist);
    betabeta=LAbeta(1:q-1,1);
    dispbeta=[0 LAbeta(q,1)]';
    LAbeta=[betabeta' dispbeta']';
  elseif dispbeta(2)==-1,
    Da1=ones(1,a(1))./dispbeta(1);
    [LAbeta its]=fminjim('MaxLa',[betabeta' dispbeta(1)]',20, ...
                  1.e-3,.0001,y,m,x1,A1,Da1,[a(1) 0],offset,dist);
    betabeta=LAbeta(1:q-1,1);
    dispbeta=[LAbeta(q,1) 0]';
    LAbeta=[betabeta' dispbeta']';
  end;
  if min(dispbeta)==-1,
    dispbeta=[0 0]';
  end;
  Labeta =MaxLa([betabeta' dispbeta']',y,m,x1,[A1 A2],D,a, ...
              offset,dist);
  Lagrid(index)=Labeta;
```

```
    [GHbeta its]=fminjim('MaxGH',[beta1' dispbeta']',20, ...
                1.e-3,.0001,y,m,x1,[A1 A2],D,a,offset,dist,GHstuff);
    betabeta=GHbeta(1:q-1,1);
    dispbeta=GHbeta(q:q+1,1);
    if dispbeta(1)==-1,
      Da2=ones(1,a(2))./dispbeta(2);
      [GHbeta its]=fminjim('MaxGH',[betabeta' dispbeta(2)]',20, ...
            1.e-3,.0001,y,m,x1,A2,Da2,[0 a(2)],offset,dist,GHstuff);
      betabeta=GHbeta(1:q-1,1);
      dispbeta=[0 GHbeta(q,1)]';
      GHbeta=[betabeta' dispbeta']';
    elseif dispbeta(2)==-1,
      Da1=ones(1,a(1))./dispbeta(1);
      [GHbeta its]=fminjim('MaxGH',[betabeta' dispbeta(1)]',20, ...
            1.e-3,.0001,y,m,x1,A1,Da1,[a(1) 0],offset,dist,GHstuff);
      betabeta=GHbeta(1:q-1,1);
      dispbeta=[GHbeta(q,1) 0]';
      GHbeta=[betabeta' dispbeta']';
    end;
    if min(dispbeta)==-1,
      dispbeta=[0 0]';
    end;
    GH2beta =MaxGH([betabeta' dispbeta']',y,m,x1,[A1 A2],D,a, ...
               offset,dist,GHstuff);
    GH2grid(index)=GH2beta;
end;
end;
LRLA=2*(min(Lagrid)-Lagrid);
LRGH2=2*(min(GH2grid)-GH2grid);
ZGLS=(betagls0-betagrid)/sqrt(glscov0);
LRGLS=-ZGLS.*ZGLS;
LRIWLS=-2*(liwls-max(liwls));
ZQL=(betahi0-betagrid)/sqrt(qlcov0);
LRQL=-ZQL.*ZQL;
```

```
save profile
```

profrun.m

```
indata
setup
x=xfull;
q=min(size(x));
optLa
x0=zeros(y);
beta0=-betaLa(2)/(2*betaLa(3));
x1=[x(:,1) x(:,2)-x(:,3)/(2*beta0) x(:,4)];
beta1=betaLa([1 2 4],:);
b1=betaLa(2);b2=betaLa(3);vb1=qlcov(2,2);
vb2=qlcov(3,3);vb12=qlcov(2,3);
qlcov0=(beta0*beta0)*(vb1/(b1*b1)+vb2/(b2*b2)-2*vb12/(b1*b2));
b1=betaLa(2);b2=betaLa(3);vb1=glscov(2,2);
vb2=glscov(3,3);vb12=glscov(2,3);
glscov0=(beta0*beta0)*(vb1/(b1*b1)+vb2/(b2*b2)-2*vb12/(b1*b2));
betagls0=-betagls(2)/(2*betagls(3));
betahi0=-betahi(2)/(2*betahi(3));
profile
quit
```

salam.m

```
salamander1
setup2
initial2
optIGLS
optREML
optLa
save salam2
quit
```

salam3pr.m

```
load /home/stat/pratt/matlab/salam2c
itime=clock;
beta0=betaLa(1);
beta1=[betaLa(2) betaLa(3) betaLa(4)]';
x0=x(:,1);
x1=[x(:,2) x(:,3) x(:,4)];
se0=sqrt(glscov(1,1));
profLA
timepr1=etime(clock,itime);
beta1grid=betagrid;
La1grid=Lagrid;
save salam2c
itime=clock;
beta0=betaLa(2);
beta1=[betaLa(1) betaLa(3) betaLa(4)]';
x0=x(:,2);
x1=[x(:,1) x(:,3) x(:,4)];
se0=sqrt(glscov(2,2));
profLA
timepr2=etime(clock,itime);
beta2grid=betagrid;
La2grid=Lagrid;
save salam2c
itime=clock;
beta0=betaLa(3);
beta1=[betaLa(1) betaLa(2) betaLa(4)]';
x0=x(:,3);
x1=[x(:,1) x(:,2) x(:,4)];
se0=sqrt(glscov(3,3));
profLA
timepr3=etime(clock,itime);
beta3grid=betagrid;
La3grid=Lagrid;
```

```
save salam2c
itime=clock;
beta0=betaLa(4);
beta1=[betaLa(1) betaLa(2) betaLa(3)]';
x0=x(:,4);
x1=[x(:,1) x(:,2) x(:,3)];
se0=sqrt(glscov(4,4));
profLA
timepr4=etime(clock,itime);
beta4grid=betagrid;
La4grid=Lagrid;
save salam2c
clear

load /home/stat/pratt/matlab/salam3c
itime=clock;
beta0=betaLa(1);
beta1=[betaLa(2) betaLa(3) betaLa(4)]';
x0=x(:,1);
x1=[x(:,2) x(:,3) x(:,4)];
se0=sqrt(glscov(1,1));
profLA
timepr1=etime(clock,itime);
beta1grid=betagrid;
La1grid=Lagrid;
save salam3c
itime=clock;
beta0=betaLa(2);
beta1=[betaLa(1) betaLa(3) betaLa(4)]';
x0=x(:,2);
x1=[x(:,1) x(:,3) x(:,4)];
se0=sqrt(glscov(2,2));
profLA
timepr2=etime(clock,itime);
beta2grid=betagrid;
```

```
La2grid=Lagrid;
save salam3c
itime=clock;
beta0=betaLa(3);
beta1=[betaLa(1) betaLa(2) betaLa(4)]';
x0=x(:,3);
x1=[x(:,1) x(:,2) x(:,4)];
se0=sqrt(glscov(3,3));
profLA
timepr3=etime(clock,itime);
beta3grid=betagrid;
La3grid=Lagrid;
save salam3c
itime=clock;
beta0=betaLa(4);
beta1=[betaLa(1) betaLa(2) betaLa(3)]';
x0=x(:,4);
x1=[x(:,1) x(:,2) x(:,3)];
se0=sqrt(glscov(4,4));
profLA
timepr4=etime(clock,itime);
beta4grid=betagrid;
La4grid=Lagrid;
save salam3c
clear

load /home/stat/pratt/matlab/salam4c
itime=clock;
beta0=betaLa(1);
beta1=[betaLa(2) betaLa(3) betaLa(4)]';
x0=x(:,1);
x1=[x(:,2) x(:,3) x(:,4)];
se0=sqrt(glscov(1,1));
profLA
timepr1=etime(clock,itime);
```

```
beta1grid=betagrid;
La1grid=Lagrid;
save salam4c
itime=clock;
beta0=betaLa(2);
beta1=[betaLa(1) betaLa(3) betaLa(4)]';
x0=x(:,2);
x1=[x(:,1) x(:,3) x(:,4)];
se0=sqrt(glscov(2,2));
profLA
timepr2=etime(clock,itime);
beta2grid=betagrid;
La2grid=Lagrid;
save salam4c
itime=clock;
beta0=betaLa(3);
beta1=[betaLa(1) betaLa(2) betaLa(4)]';
x0=x(:,3);
x1=[x(:,1) x(:,2) x(:,4)];
se0=sqrt(glscov(3,3));
profLA
timepr3=etime(clock,itime);
beta3grid=betagrid;
La3grid=Lagrid;
save salam4c
itime=clock;
beta0=betaLa(4);
beta1=[betaLa(1) betaLa(2) betaLa(3)]';
x0=x(:,4);
x1=[x(:,1) x(:,2) x(:,3)];
se0=sqrt(glscov(4,4));
profLA
timepr4=etime(clock,itime);
beta4grid=betagrid;
La4grid=Lagrid;
```

```
save salam4c
clear
quit
```

H. ROUTINES TO CONDUCT SIMULATIONS

samples.m

```
% Matlab script file: samples.m
% Generates random samples of Poisson or Binimial with normal
% random effects added in the link. Uses the following
% predefined variabes:
% str1,str2,str3 -- seeds to cause 3 random number streams
% dist -- indicator of Poisson (=1) or binomial (=2);
% A1 and A2 -- design matrices for random effects, putting 2
%    nested in 1.
% y,x,m -- data and fixed effects desing matrix for data.
% tau1,tau2 -- desired variance of random effects.
% beta -- desired fixed effects value.
Idot=a(1);  % number of levels in 1st nesting.
Jdot=a(2);  % sum of levels of 2nd nesting, summed over 1st levels

% will first get the normal random effects vectors.
rand('seed',str1);          %set stream 1's seed to str1.
rand('normal');             %set distribution to normal.
U=sqrt(tau1)*rand(Idot,1); %vector of normal(0,tau1) of length Idot.
str1=rand('seed');          %save stream's new seed.
rand('seed',str2);          %set stream 2's seed to str2.
V=sqrt(tau2)*rand(Jdot,1); %vector of normal(0,tau2) of length Jdot.
str2=rand('seed');          %save stream's new seed.

% find sampled canonical parameter, theta.
theta=x*beta+A1*U+A2*V;

% now compute new sample Y from desired distribution with mean
% at g(theta).
rand('seed',str3);
rand('uniform');
```

```
if dist==1,
    crit=exp(-exp(theta));
    R=rand(length(y),round(exp(max(theta))));
    size(R)
    S=R(:,1);
    check=S>=crit;
    Y=check;
    index=1;
    while any(check),
    index=index+1;
    if index==round(exp(max(theta))),
       index=1
       R=rand(length(y),round(exp(max(theta))));
    end
    S=S.*R(:,index);
    check=S>=crit;
    Y=Y+check;
end
else
   R=rand(length(y),max(m));
   mu=exp(theta)./(1+exp(theta));
   p=kron(ones(1,max(m)),mu);
   S=R<=p;
   if max(m)==min(m),
     Y=sum(S')';
   else,
     for index=1:length(y)
        Y(index,1)=S(index,:)*[ones(1,m(index))  ...
            zeros(1,max(m)-m(index))]';
     end
   end;
end
clear U V Jdot Idot R S check crit index mu p theta;
```

current.m

```
    % Matlab script file current.m    updated Feb 24, 1992
indata
setup
str1=38991;
str2=46901;
str3=10456;
tau1=.25;
tau2=.1;
disppar=[.25 .1]';
if dist==1,
  beta=3;
else,
  beta=1;
end;
numsam=1;
glim=zeros(numsam,3);
ql=zeros(numsam,2);
gls=zeros(numsam,4);
Las=zeros(numsam,9);
GH2=zeros(numsam,7);
GH5=zeros(numsam,7);
GHwts
for i=1:numsam,
  samples
  y=Y;
  opt
disp('opt done')
   lbetahi=y'*x*betahi-sum(B1(x*betahi,m,dist));
   lbeta=y'*x*beta-sum(B1(x*beta,m,dist));
   glim(i,:)=[betahi lbetahi lbeta];
   [mu V]=Bw(x*betahi,m,dist);
   W=diag(V);
   ql(i,:)=[oversig (betahi-beta)/sqrt(oversig*inv(x'*W*x))];
```

```
    [mu V]=Bw(x*betagls,m,dist);
    D=dispgls(1)*A1*A1'+dispgls(2)*A2*A2';
    Wgls=inv(inv(diag(V))+D);
    gls(i,:)=[betagls dispgls' (betagls-beta)/sqrt(inv(x'*Wgls*x))];
    Las(i,:)=[betaLa dispLa dispbeta' LaMax Labeta GHLMax GHLbeta];
    GH2(i,:)=[betaGH2 dispGH2 dispGH2b'GH2Max GH2beta];
    GH5(i,:)=[betaGH5 dispGH5 dispGH5b'GH5Max GH5beta];
end;
endtime=clock;
save current endtime beta tau1 tau2 glim ql gls Las GH2 GH5;
quit;
```