September 24, 2007

DRAFT DEVELOPMENT AND TESTING REPORT

FOR

SOFTWARE MODULES TO INCORPORATE BACKGROUND DOSE/RESPONSE IN
QUANTAL MODELS WITHIN BENCHMARK DOSE MODELING SOFTWARE (BMDS)

Prepared by
Sowmi Utiramerur
Douglas Mooney
(Edited by John Fox, EPA Work Assignment Manager)

BATTELLE
505 King Avenue
Columbus, OH 43201-2693

EPA Disclaimer

This document is a draft for review purposes only and does not constitute U.S. Environmental Protection Agency Policy. Mention of trade names and commercial products does not constitute endorsement or recommendation for use.

Battelle Disclaimer

This report is an interim work prepared by the United States Government by Battelle and is for discussion purposes only. In no event shall either the United States Government or Battelle have any responsibility or liability for any consequences of any use, misuse, inability to use, or reliance upon the information contained herein, nor does either warrant or otherwise represent in any way the accuracy, adequacy, efficacy, or applicability of the contents hereof.

Table of Contents

SOFTWARE MODULES TO INCORPORATE BACKGROUND DOSE/RESPONSE IN QUANTAL MODELS WITHIN BENCHMARK DOSE MODELING SOFTWARE (BMDS)

DEVELOPMENT AND TESTING REPORT

1.0   INTRODUCTION AND OBJECTIVES

The U.S. Environmental Protection Agency (EPA)'s Benchmark Dose Software (BMDS) is a tool to facilitate the application of benchmark dose (BMD) methods to EPA's chemical risk assessments. EPA uses BMD methods to estimate reference doses (RfDs) and reference concentrations (RfCs) for noncancer human health effects. These methods involve fitting mathematical models to dose-response data and using the results to select a BMD that is associated with a predetermined benchmark response (BMR), such as a 10% increase in the incidence of a particular lesion or a 10% decrease in body weight gain.

Table 1 lists the quantal models included in EPA's current version of BMDS. Some of these models (i.e., multistage, Cancer, log-logistic, log-probit, Weibull, and gamma models) include a background *response* parameter γ. This parameter represents the effect that various background contributors may have on the response and is independent of the dose of the chemical under study. EPA required alternative versions of these models, in which the background parameter was re-defined to be additive to the effect of the dose rather to the response:

Model with background response parameter, γ: $P(\boldsymbol{\beta}, \gamma, x) = \gamma + (1-\gamma)*F\{\boldsymbol{\beta}, x\}$

Model with background dose parameter, η: $P(\boldsymbol{\beta}, \eta, x) = F\{\boldsymbol{\beta}, (x + \eta)\}$

Therefore, in the background dose versions of these models, the parameter γ is dropped, and the parameter η is added, resulting in no net change to the total number of parameters in the model.

Table 1 also shows that the logistic and probit models (*without* log-transformation of the dose) currently within BMDS implicitly accounts for a background dose effect, even though a background dose parameter is not explicitly estimated. However, these models do not feature a background *response* parameter. Thus, EPA also required alternative versions of these models to add an explicit background response parameter γ to the logistic and probit models. In doing so, the total number of parameters in each of these two models increases from two to three.

Table 1.   Comparison of current BMDS quantal models with *new models* allowing a background dose or background response parameter

Model Name[1]	Functional Form of the Model	Parameters Added	Parameters Dropped	#
Multistage	$\gamma + (1-\gamma) [1 - \exp\{- \sum_{i=1}^{k} \beta_j X^{\,j} \}]$			1+k
multistage_bgd	$1 - \exp\{- \sum_{i=1}^{k} \beta_j (X + \eta)^{\,j} \}$	η	γ	1+k
Logistic	$[1 + \exp\{- (\alpha + \beta X)\}]^{-1}$			2
logistic_bgr	$\gamma + (1-\gamma) [1 + \exp\{-(\alpha + \beta X)\}]^{-1}$	γ	None	3
Probit	$\Phi\{ \alpha + \beta X \}$			2
probit_bgr	$\gamma + (1-\gamma) \Phi\{ \alpha + \beta X \}$	γ	None	3
Log_logistic	$\gamma + (1-\gamma) [1 + \exp\{- (\alpha + \beta \log\{X\})\}]^{-1}$			3
log_logistic_bgd	$[1 + \exp\{- (\alpha + \beta \log\{ X + \eta \})\}]^{-1}$	η	γ	3
Log_probit	$\gamma + (1-\gamma) \Phi\{ \alpha + \beta \log\{X\} \}$			3
log_probit_bgd	$\Phi\{ \alpha + \beta \log\{X + \eta \} \}$	η	γ	3
Gamma	$\gamma + (1-\gamma) [\Gamma(\alpha)^{-1} \int_0^{\beta X} t^{\alpha-1} e^t dt]$			3
gamma_bgd	$G(\beta(d+\eta), \alpha) / \Gamma(\alpha)$ G is the incomplete gamma function	η	γ	3
Weibull	$\gamma + (1-\gamma) [1 - \exp\{-\beta X^{\alpha}\}]$			3
Weibull_bgd	$[1 - \exp\{-\beta (X + \eta)^{\alpha}\}]$	η	γ	3

[1] Names in regular type denote modules (i.e., *.exe files) that currently exist within BMDS, and they specify the model form. Names in italics denote modules that are newly created for BMDS and represent alternative forms of the models with background parameter included. Note that because of unresolved convergence issues, the alternative form of the log-logistic model is not being released for use in BMDS at this time. The Cancer module is not separately tabulated; it a version of the multistage module that requires all parameters to be non-negative and reports the "cancer slope factor" which is the ratio BMR/BMDL.

This report details the design approach, testing, and validation methodology used in the development of the following additional capability for the BMDS application:

    a. Developing alternative versions of the multistage, Cancer, log-logistic, log-probit, gamma, and Weibull models that incorporate a background dose parameter (η) in place of a background response parameter (γ).

    b. Developing alternative versions of the logistic and probit models that incorporate an explicit background response parameter (γ).

    c. Computing an upper confidence bound on the BMD (BMDU) under both the current and alternative models, which will accompany the lower confidence bound (BMDL) that the BMDS already calculates within the current models. (Currently, calculating the BMDU is not critical to EPA's quantitative risk assessment methods.)

In adding new modules that correspond to alternative versions of the quantal models, the computer code associated with modules included in the current version of BMDS was modified to the least extent possible. However, significant changes were necessary to the code that performs certain mathematical operations, such as calculating the likelihood function and (for the multistage, Cancer, and gamma models) the Fisher Information Matrix. A list of changes made to the BMDS file set as a result of this effort is given in Appendix C.

While supporting detail is provided in later sections of this report, the following general conclusions were drawn across all five quantal models to which a background dose parameter was added:

- These quantal models with both background dose and dose response parameters were found to be poorly suited to fit dose-response data that included a high percentage (> 50%) of positive responses within the control dose group. Either the models produced poor fits, or sometimes the *donlp3* optimizer failed to converge on a minimum. It should be noted that such high positive response (>50%) within the control dose is not typical of dose-response data used in risk assessment. Few carcinogen bioassay datasets increase from a high (>40%) control response level, and in such cases, the analyses must be interpreted with caution.

Additionally, there were issues with obtaining convergence when calculating the BMDU for the Weibull model with background dose using both SAS® and BMDS. The BMDS module for this alternative model form currently converges for the test cases considered to date, and the reported solutions match those reported by SAS® software in the few cases that SAS®-based programs achieved convergence upon their execution. Furthermore, SAS®-generated results are inconsistent with respect to the optimization approach used. Thus, while no known problems exist with computing the BMDU under

the Weibull model with background dose parameter, it is the least validated of the new model calculations. More details are given in Section 4.7.

2.0   DESIGN APPROACH

Instead of modifying existing modules, EPA developed new modules (starting with a copy of existing code) to implement the new versions of quantal modules. From the BMDS interface, users will be able to select either the current or the new (alternative) version of a model (as noted in Table 1), which will then determine the appropriate module to call. For example, when applying the multistage model, users will have a choice of "multistage" (i.e., the current version with background response parameter) or "multistage_bgd" (i.e., the alternative version with background dose parameter).

EPA has developed new modules, distinct from current modules, which implement the alternative forms of the quantal models. This was done for three principal reasons:

- To separate code changes necessary for the alternative models from those changes that may be occurring within other modifications to current versions of the BMDS modules, thereby avoiding the need to reconcile and integrate changes between the two code sets.

- To isolate existing code, thereby minimizing any unintended effects on coding that may influence existing results.

- To implement a modular approach to organizing the revised code, due to the considerable modification of existing code and the desire to make the code consistent with existing BMDS architecture.

Initially, when planning the development of revised modules, EPA expected that each model would include both background dose and background response parameters. However, preliminary analysis indicated that incorporating both parameters simultaneously within a model would lead to several significant numerical complications. For example, consider a version of the multistage model with both types of background parameters included in the model. If the user wished to restrict one of the parameters to zero, the restricted model would be equivalent to a model with a single background parameter. When this scenario was tested, the restricted two-parameter model was less stable and had poorer convergence compared to the equivalent single-parameter version of the model. It became clear that a considerable amount of time and effort would be needed for development, coding, and testing of modules that included both types of background parameters within their respective models. Thus, EPA determined that a particular module would feature a model with only one type of background parameter, and that separate modules would be prepared for different forms of the model (i.e., background dose versus background response parameters).

Likelihood Method. As is used in the current versions of the models, the profile likelihood method was used for estimating the BMDL and BMDU in all new modules. For the multistage/Cancer and gamma models, differentiation of the likelihood function was used to compute the Fisher Information Matrix, which is an intermediate step in

computing a covariance matrix. Otherwise, for the other models, a finite difference method was used to differentiate the likelihood function.

Functional Requirements. The software development effort adhered to the following functional requirements:

- The format of the input file (.d) was modified to allow the user to specify the background dose parameter.
- The output generated by executing the modules was modified to report that a background dose model was used.
- Development of new modules for the alternative version of a given model began by modifying a copy of the code for the existing version of the model. This code often consisted of a mixture of FORTRAN and ANSI C code. All new modules were written in ANSI C, and any existing FORTRAN source code was rewritten in ANSI C.
- New modules have quality mechanisms such as error trapping in place. Although existing BMDS modules were not modified, any problems and issues that may have been encountered with existing BMDS code were reported to the EPA Work Assignment Manager (WAM). Any such observations would be *ad-hoc* in nature and not the result of a concerted quality review.
- New modules corresponding to alternative models with a background dose parameter included were named *modelname_bg_dose.c*
- All code was developed and tested using the Microsoft Windows XP Service Pack operating system.
- Newly written code was compiled using the version of the GCC complier provided in the MinGW5.0 bundle.
- Programs featuring the new code were designed and tested under the assumption that it would be executed only in a MS Windows XP Service Pack 2 environment.
- Regression model testing was done to assure that code modifications did not adversely affect existing functionality.

2.1   Differences in Modules between Current and Alternative Model Versions

For a given quantal model, the principal differences between the new module prepared for the alternative version and the existing module within BMDS for the current version are as follows:

Optimizer The approach currently featured within BMDS was used to compute maximum likelihood estimates (MLEs) of model parameters including the additional background parameter. This approach used the ANSI C version of the *donlp2* optimizer known as *donlp3*, which features dynamic memory allocation (donlp2_intv_dyn)[1]. Due to differences in how ANSI C implements arrays compared to FORTRAN, attempts to

---

[1] This terminology is consistent with informal usage on the website containing the *donlp2* software at plato.la.asu.edu/donlp2.html

transfer operations between C and FORTRAN are prone to conversion errors. In addition, use of the C version improves code readability and maintainability.

Confidence Interval for BMD. Currently within BMDS, a parameter substitution method is used to compute a lower confidence limit (BMDL) on the estimated BMD and, for the Cancer model, to compute the BMDU as well. This method involves substituting the BMD as a parameter in the log likelihood function, in place of another parameter which was determined to have the least impact on the shape of the functional space. Thus, the revised log likelihood function has the same total number of parameters as before, except BMD is now included as a parameter. For example, consider the formula for calculating added risk[2] associated with the BMD under the log-logistic model:

$$BMR = \pi(BMD) - \pi(0)$$

$$= \gamma + (1-\gamma) [1 + \exp\{-(\alpha + \beta \ln(BMD))\}]^{-1} - \gamma$$

$$= (1-\gamma) [1 + \exp\{-(\alpha + \beta \ln(BMD))\}]^{-1}$$

where $\pi(d)$ denotes the probability of adverse effect at dose d. Rearranging terms to express the intercept parameter α as a function of BMD gives

$$\alpha = -\ln[(1-\gamma)/BMR -1] - \beta \ \ln(BMD). \quad (1)$$

Thus, because α can be expressed as a function of β, γ and BMD, the log likelihood function can be expressed in terms of only these parameters as well. To see this for the logistic example note that the probability of response at a given dose is

$$p(dose; \alpha, \beta, \gamma) = \gamma + (1-\gamma) [1 + \exp\{-(\alpha + \beta \ln(dose))\}]^{-1}. \quad (2)$$

Making the parameter substitution of (1) into (2) yields

$$p(dose; BMD, BMR, \beta, \gamma) =$$

$$\gamma + (1-\gamma) [1 + \exp\{-(-\ln[(1-\gamma)/BMR -1] - \beta \ \ln(BMD) + \beta \ln(dose))\}]^{-1}. \quad (3)$$

Because the likelihood function for a dichotomous model has the form

$$L = \sum L_i (N_i, n_i, dose_i; \Theta) \quad (4)$$

where

$$L_i = n_i \ln(p_i) + (N_i - n_i) \ln(1 - p_i), \quad i = 1, 2, \cdots, k \quad (5)$$

---

[2] Added risk is defined as the calculated difference in risk of a particular condition between those who are exposed and those who are not.

and Θ is the vector of model parameters, it is straightforward to see that substituting (3) at each of the k observed doses into (5), yields a likelihood function (4) with $\Theta = $ (BMD, BMR, β, γ).

Also, BMDL can be estimated by minimizing this function with respect to BMD. However, because the intercept parameter is eliminated from the model, the estimate of BMDL might not correspond to the true minimum that satisfies the necessary constraints. The effect of the parameter substitution on the minimization of BMD can be unpredictable and is highly dependent on the shape of the functional space. The revised (background dose) modules discussed in this report use the BMD as an additional parameter, with a constrained minimization on the BMD parameter. Even though in test cases there were no significant differences (less than 1%) in the results using parameter substitution, design changes have been made to diminish potential convergence issues. The computation of the BMDU was performed in analogous manner with similar code developed under this effort.

Fisher Information Matrix (FIM) For the multistage/Cancer and gamma models, differentiation of the profile likelihood function was used to compute the FIM, an intermediate step in computing a covariance matrix. For the other models, a finite difference method, used in existing BMDS modules, was used to differentiate the profile likelihood function. Small imprecision in the calculation of parameter values, along with a numerical (Finite Difference) technique to computing the second differential of the log-likelihood, leads to increased imprecision in the FIM. For the gamma model, numerical approximations were made while calculating the second order differential of the objective function, and additional approximations resulted when calculating the inverse of the FIM. To mitigate this potential problem, the FIM was calculated using analytical methods to differentiate the objective function based on series expansion [Moore, 1982] or continued fraction expansion of the Incomplete gamma Integral.

Initial Parameter Estimates: The initial estimates of parameters in BMDS are calculated by linear regression on a re-parameterized objective function. The addition of the background dose or response parameter does not allow the objective function to be re-parameterized into a linear form that yields a general algorithm that can be coded for most models.

Some special transformations and techniques like Taylor's theorem approximations were used with the Weibull and gamma models (see details in their respective sections below). For these models, the initial estimate of the background dose or response parameter was set to zero with the exception of the multistage model for which consistent convergence was seen with an initial background response parameter estimate of 0.5. (This was not an issue with the Cancer model due to the constraint of positive betas.) No convergence issues were encountered within the test cases developed for validation based on these initial parameter choices.

To evaluate whether choice of initial parameter estimates led to the finding of a local rather than global maxima and whether a grid of starting values might be beneficial,

studies were conducted on a number of test cases using grid searches. These studies found that for all of the different sets of initial conditions that led to successful convergence, the same set of maximum values was obtained in every case.

Scaling of Parameters   To avoid convergence problems when fitting the multistage and Cancer models, the dose values, and hence the beta parameters (i.e., coefficients of dose), are scaled by the maximum dose value. Specifically, if $max(X)$ denotes the maximum dose value X, then the jth stage of the multistage model, $\beta_j X^j$, is expressed equivalently as $[max(X)]^j \cdot \beta_j \cdot (X/max(X))^j$ when the model is fitted. Naturally the estimates of the β_j terms are unscaled prior to reporting their values in the output file. Because the argument involving dose is a polynomial function, dividing the dose values by the maximum dose in the original multistage model (i.e., with background response parameter) ensures that powers of numbers less than or equal to one are taken within the polynomial function, eliminating the potential for very large numbers to cause numerical computation problems. However, the addition of a background dose parameter η without restriction does not ensure that the argument $(X + \eta)$ will be less than one. This is a concern in the multistage, gamma, and Weibull models, because this argument is raised to a power, e.g., $\beta_j (X + \eta)^j$. Despite this concern, no convergence issues due to this issue were encountered during testing. For example, the maximum values of the background dose parameter encountered during testing of the multistage model were less than five. It should be noted that some convergence issues might occur if the background dose parameter values are on the order of the maximum dose. Although code has been implemented to restrict background dose for the multistage and Cancer models, it is not currently functional as it requires a change to the input (*.d) file format (Appendix B). In the future, if convergence issues are encountered, the restriction on the background dose parameter can be activated. Although scaling of dose coefficients is not needed for the other quantal models, it is featured in the current version of BMDS, and therefore, was retained (although its motivation is not certain).

3.0   SOFTWARE QC TESTING PROCEDURES

Software quality control (QC) activities involved both validation (i.e., testing) and verification (i.e., review for completeness and adherence to process) of procedures and included the types of testing specified in EPA's draft guidance from the Council for Regulatory Environmental Modeling [USEPA, 2003]. These types of procedures are discussed in this section.

3.1   Validation Procedures

The primary goal of validation efforts was to ensure that the written and compiled code produced correct results and otherwise operates correctly. This section describes the steps taken to achieve this goal.

For each new module addressed by this testing document, the coding process was preceded by the preparation and approval of a Plan and Approach document. The Plan and Approach document contained validation cases that ideally would be used to compare results of executing the enhanced BMDS software to results reported by independent sources such as peer-reviewed articles or trusted software that performs similar calculations. Because no published examples of results for these models were found, comparisons were made to results generated by third party mathematical and statistical software packages, in this case, SAS® and *Mathematica*®. The Plan and Approach document also specified any anticipated implementation problems. As coding proceeded, any problems were brought to EPA's attention, and additional test cases were developed and added to the test suite. Collectively, these formed a Testing Plan which consisted of a series of cases in a test framework and a set of *.(d) files that were exercised using *.bat files.

The test cases featured in the Testing Plan were constructed to ensure that newly written code performs calculations correctly, is fully accessible, and achieves all of its objectives when incorporated within the BMDS. They also were used to test the functionality currently existing in the BMDS and to verify that the newly developed functionality does not adversely affect existing functionality. Specifically, these test cases provided before-and-after tests to verify that representative BMDS functionalities give the same results after the introduction of the newly developed features. These tests do not verify that functionality currently existing in the BMDS produces correct output or behaves as expected under a full range of allowable inputs.

Unit testing refers to the testing of individual functions or a module of related functions. Developed by staff members responsible for each unit and individual module (e.g., statisticians and code developers) unit tests addressed each functional unit of code for which such a test was relevant, as well as each module as a whole. In newly written code, unit tests were performed on both computations and operations such as file opening. Since existing BMDS code was reused wherever possible, unit testing was limited to computations in the newly developed components. Unit tests were performed

on maximum likelihood calculations, parameter estimates, covariance matrix computations, and computations of BMD, BMDL, and BMDU. In general, where units represent mathematical or statistical functions, the test results, when possible, were compared against published results. When this was not feasible, either hand calculations or results from generally trusted analytical software packages (i.e., SAS® and *Mathematica*®) were used. Team statisticians produced sets of data to test the ability of each module to produce correct results. Unit tests were placed in a framework for reuse (given the name *test_<module_name>.c*) and were provided with the code package.   Before any piece of code is released, it must pass all established unit tests.

Along with unit testing, *system integration tests* were performed on the completed modules to test the integration of multiple units. These tests verified that modules work correctly when exercised through the BMDS shell[3], that numerical results are correct, and that the module fully integrates with the shell (i.e., all expected functionally is present, all expected output is created, and no errors are produced). Test.bat files, provided with each module, automatically execute these system integration cases.

Many test cases were commonly used for both system integration and unit testing. Unit testing was done in a C-based software framework that tested individual function calls, while system integration tests used *.(d) files for input and were invoked by calling BMDS from the command line. Each unit test incorporated established passing criteria in the framework, and its output was presented in a Pass/Fail report with numerical results. System integration tests were used for regression testing (i.e., testing that future version of code do not change existing calculations), with standard BMDS output. Calculations must be checked manually, although some system integration tests addressed general program operation and error trapping.

Additional detail follows on how SAS® and *Mathematica*® were used to develop the test cases.

3.1.1   SAS® 9.1.3

The SAS® System is an independent third-party statistical software tool used to generate results for test cases. SAS® code used in Wheeler (2005) to fit traditional quantal models was available for this effort, serving as a basis for developing verification cases of the new quantal models. Wheeler's programs use PROC NLMIXED to perform the optimization step for both parameter fitting and lower bound estimation. The rest of the programs are data steps and macros. Code obtained from Wheeler (2005) for the Weibull model, and the validation code for the background dose/background response models described in this document, are listed in Appendix

---

[3] Here, the term "shell" refers to the system of BMDS programs underlying, and invoked by, the graphical user interface (e.g., "multistage.c") and all the programs which support it (including those in the Assist folder). Testing did not involve the graphical user interface. Instead, the DOS batch files executed commands like "multistage.c testdata1.(d)", which exercised the usual estimation and reporting actions and produced the usual BMDS results files, e.g., "testdata1.OUT", "testdata1.002", and "testdata1.plt".

E. MLEs of background and other parameters, the covariance matrix, and the BMD were generated for each test case. Because the fitting algorithms of SAS® are independent from the algorithms developed for implementing the alternative models within BMDS, they were used to validate numerical results from the BMDS algorithms in C. *Mathematica*® was used whenever convergence issues were encountered within SAS® when performing constrained optimization to compute BMDL and BMDU. While some SAS® code executions led to convergence for some test data sets, problems were encountered with other test data sets which led to the development of a limited set of validation cases when SAS® was used.

3.1.2 *Mathematica*® 5.2

Mathematica® is a high-level language that allows for straightforward coding of mathematical algorithms using normal mathematical notation. It contains many built-in functions for numerical tasks such as root finding and optimization and has capabilities for performing very complicated optimizations involving nonlinear response functions over nonlinear constraints that are often difficult to find in commercial "off-the-shelf" analysis packages.

When using *Mathematica*® in validation efforts, optimizer routines were found to be quite inconsistent and highly dependent on the initial specified conditions. In order to get the optimizer to converge properly, parameter sets that were close to the results estimated by BMDS were input into *Mathematica*®. Although this approach can reduce the usefulness of the results as a validation tool, no suitable approach exists that will generate unbiased results that could be used to validate the calculations of BMDL and BMDU. Furthermore, because the same general approach used to develop the BMDS code was used to write *Mathematica*® code, this validation technique helps identify coding errors but does not validate methodology or approach.

3.2   Verification Procedures

During the review stage, the QA review performed on software modules included verification procedures such as the following:

1. All software development products, including requirements and specifications, designs, code, test plans/cases, documentation, and examples were peer reviewed by at least one individual.

2. Every product was reviewed for completeness, clarity, accuracy, precision, and consistency; and code was reviewed for adherence to specifications, algorithm selection, adherence to standards, and error handling.

3. Test plans and test cases were reviewed for correctness and thoroughness in exercising the code.

Two reviewers who were not involved in the development process reviewed the code according to these guidelines and completed a checklist of test findings. Any identified deficiencies were corrected.

3.3   QA Processes

The coding and testing process adhered to the following QA processes:

- Source Control. Source code was maintained in a source safe, namely the MKS Source Integrity System.

- Unit and System Integration Testing. As discussed earlier, a testing framework was established to test individual calculations and function calls (units). For system integration testing, a set of *.(d) file and batch files that run BMDS modules were defined and implemented.

- Regression Tests. The unit test suites served as regression tests. That is, if future modifications are made to the code set, the unit tests can be re-executed to identify changes in output due to the code revisions. The programs that conduct unit tests (named *test_<module_name>.c*) are included in the code package.

- Tests of Impact on Existing BMDS. The quantal datasets that accompanied the current version of BMDS (version BMDS-1-4-20051110) were executed both before and after the additional capability was incorporated into the software. Specifically, the current version of the quantal modules was executed prior to adding the new code to the Assist directory and the new code modules to the BMDS folder, and then again after installation of the new code. The output files were compared and found to be identical, indicating that the presence of the new modules and changes to the Assist folder did not impact existing module performance.

- Code Review. The code to calculate the BMDL and BMDU was initially developed by a software programmer and was carefully examined and revised by a mathematician/statistician in the course of developing verification and validation tests.

3.4   Numerical Precision

Prior to implementing the validation process, it was necessary to determine acceptable bounds and tolerances within which an estimate needed to fall. This is intrinsically a mathematical issue. While BMDS allows the user to specify a tolerance for "parameter estimates" (default: 1.49012E-8), this actually represents a tolerance on the constrained MLE and not on the model parameters at which that maximum occurs. Establishing tolerances for parameter estimates is much more difficult, because the exact maximum or an objective function cannot be determined. Rather than finding a

single, mathematically correct solution, searches occur until one of many numerical solutions is identified as being within some acceptable tolerance.

Suppose that the objective function is estimated to within some acceptable tolerance. This corresponding tolerance interval translates to a region in the parameter space over which optimization is performed. The extent of the region depends on the geometry (curvature and gradients) of the objective surface. This region can be large if the objective surface is relatively flat in one or more parameter directions. In this case, there could be a large region in the parameter space satisfying even very tight bounds on the objective function. Chapter 3 of Seber and Wild (1989) discuss the issues of ill-conditioned problems and curvature effects. They do note, however, that for well-conditioned problems, the magnitude of the error on parameters can be bounded by

$$\left\| \bar{\theta}^{(a)} - \bar{\theta}^{(a-1)} \right\| < \sqrt{\tau_h} \bullet \left(1 + \left\| \bar{\theta}^{(a)} \right\|\right)$$

where $\vec{\theta}^{(a)}$ is the value of the parameter vector at the ath iteration and τ_h is the relative accuracy required in the computed value of the objective function.

BMDS reports the estimated correlation[4] matrix for the parameter estimates to two decimal places. While the computations differed from those generated using SAS® by about five to ten percent under the gamma model and less than five percent under all other models, they were not consistently more or less conservative than the SAS® estimates. For example, in one test case, the estimated variance on the slope parameter was nearly ten percent more conservative than that estimated by SAS®, while the estimated variance on the power parameter was ten percent greater than the SAS®-reported result. These discrepancies are largely due to approximations made while calculating the second order differentials of the objective function and are further propagated while taking the matrix inverse on the FIM when calculating the covariance matrix. Because it could not be determined whether SAS® uses a numerical or analytical method to compute the covariance matrix, it was not feasible to determine whether SAS® or BMDS produces a more precise result. Details about covariance matrix calculation for the gamma model are given in Section 4.

3.5   Compiler-Related Issues

The MinGW5.0 compiler and the options in the Makefile for the BMDS-1-4-20051110 version of BMDS were used to compile new code associated with the alternative quantal models. However, on a recent effort to augment the BMDS modules for fitting the multistage and logistic models by estimating the slope of the fitted dose-response curve, considerable effort was made to resolve subtle issues regarding the compiler. Some of these issues are also relevant to the additional capability being discussed in this report, and therefore, are repeated for completeness. While these efforts centered on understanding the nature of differences, they did not examine which compiler and compiler options were optimal.

---

[4] BMDS reports (prints) only the correlation matrix, which is computed after estimation of the covariance matrix.

The binary representation of decimal numbers by the gcc compiler can be affected by the specific compiler version as well as how compiler flags have been set. In simple calculations, this can lead to differences in about the fifteenth decimal place with a double precision number. This is not a problem for reporting double precision numbers, as the validity of such numbers is determined only to 14 decimal places or so. However, the effect may be significant enough to lead to different optimization routes (i.e., the same code on different compilers may require a different number of iterations in *donlp3* to converge to a solution). In some cases, very slight differences in the value of the likelihood function may lead to significant differences in parameter estimates and slope as discussed elsewhere in this report (e.g., Section 3.4).

Three specific compiler-related issues uncovered during the testing process, and how they were resolved, are as follows:

1. *Different implementations of the same compiler version can produce different results.* When the mingw32 port of gcc-2.95.2 was used to compile the same code on two different machines running Windows XP SP2, reported results were not identical between the two machines upon executing the compiled code. Although the differences were very small (e.g., to the fifteenth decimal place or higher), one should expect that executing identical code on two machines but compiled with the same compiler should yield identical results. The issue was traced to one machine having the dll implementation of gcc-2.95.2 installed and the other machine having the msvcrt implementation of gcc-2.95.2 installed. When both machines used the same implementation, identical results were achieved. Therefore, a decision was made to standardize on the msvcrt version, in order to be consistent with the compilation instructions for BMDS version 1.3.2. (This issue became moot when the code was executed with version BMDS-1-4-20051110 of the BMDS software, which was compiled using MinGW5.0.)

2. *Keeping compiler options consistent.* During the validation of code that involved use of wrapper functions, differences were observed in reported results (similar in size to those encountered in issue #1) when a function was called directly versus when it was called through wrapper functions. This issue was resolved upon realizing that the Makefiles did not specify the same set of compiler options between the two situations. Standardizing the compiler options to that used by the existing BMDS code eliminated the issue.

3. *Impact of different compiler versions.* As noted earlier, coding efforts began with BMDS Version 1.3.2 and the gcc-2.95.2 compiler, but an upgrade was later made to a beta version of BMDS-1-4-20051110, which was compiled under more current versions of gcc (i.e., MinGW5.0 and the associated version of gcc). As issues #1 and #2 were being resolved, some intermediate compiler versions were investigated. It was determined that different compiler versions may give slightly different results. When calculations involve a few simple operations, these differences are within the range of valid operation for double precision numbers (i.e., less than 10^{-14}).

However, during extended calculations, such as estimating BMDL and BMDU, these differences translate into much larger differences (i.e., 10^{-7}). While the difference in computation was within double precision bounds when computing the constrained maximum likelihood, greater differences were observed in the parameters associated with that likelihood. This affected the calculation of the confidence interval boundary. Therefore, acceptance thresholds were adjusted when migrating to the MinGW5.0 compiler.

3.6   Issues Involving *donlp3*

During the development stage, one of the test cases (involving the multistage model) caused routines involving *donlp3* to suspend execution, presumably falling within an infinite loop. The issue occurred as a result of the model function returning a result that was not a number (e.g., negative infinity). Further investigation revealed that as *donlp3* searched for the optimum of the multistage likelihood function in the situation where the beta parameters were not constrained to be positive, cases occurred where the betas became large and negative. As a result, the argument to the exponential function exceeded its valid range, and an infinite number was returned. The arguments of the exponential functions in the multistage log-likelihood function must be negative for the log-likelihood function to be valid. Therefore, rather than edit *donlp3*, the multistage likelihood function was revised to return a very large negative number, namely -10^{20}, whenever the argument of an exponential was positive. Therefore, this was treated as a boundary violation issue.

In practice, for purposes of determining if the argument of an exponential was positive for this purpose, we tested whether the argument was larger than a 0.01 threshold rather than zero because it performed better for a test case that was considered at the time. This threshold was chosen in an *ad hoc* manner and may be revised based on future investigation.

3.7   Safe Exponential and Log Functions

In testing performed on modules that fit the log-logistic model, routines involving *donlp3* occasionally suspended execution. This occurred when combinations of dose and background dose values led the exponential function (*exp*) to return a value of infinity while evaluating the objective function. To overcome this problem, a "safe" exponential function was introduced which reported a value of 1E20 whenever the *exp* function in the C math library returned either a higher value or a value of infinity. This approximation did not affect the results of the MLEs or the BMDL calculations in the validation tests. While this simple solution is not necessarily ideal for any numerical problem, it works well because of the functional forms involved. As noted in Table 1, the exponential function of the log-logistic model enters in the denominator of the probability function.   Thus a value of 1E20 for the exponential makes the probability very near zero. When computing maximum likelihoods, this outcome results in highly penalized likelihoods when the logarithm of a very small number is taken and the search moves away from this region in favor of regions with higher likelihoods.

The existing BMDS software has a similar "safe" logarithmic function which uses a polynomial approximation over part of the domain. A second version of this "safe log" function was developed which replaces very small values with (-1)E20 to avoid occurrences of negative infinity entering into calculations that would cause *donlp3* to suspend execution. Testing of these two versions found the second version to perform better for the Cancer and multistage models, while the first version performed better in all other models. Therefore, both versions are in current use within the code.

4.0   TESTING INDIVIDUAL QUANTAL MODELS

Modules associated with alternative versions of individual quantal models within BMDS (noted by italics in Table 1), including the Cancer model, were tested using simulated (artificial) data. Results generated by these alternative models were validated against results generated using third party mathematical and statistical packages (i.e., SAS® or *Mathematica*®) when applied to the same test data. The verified quantities included:

- MLEs for all parameters in the respective models;
- BMD estimates;
- BMDL estimates;
- BMDU estimates; and
- Elements of the covariance/correlation matrix.

Estimates of log-likelihood, goodness of fit and analysis of deviance were not verified, because the code used in their calculation was considered legacy code (i.e., part of the existing version of BMDS) and was not modified as part of this effort.

In all eight models tested (i.e., the seven alternative forms of the models in Table 1 plus the Cancer model), each estimate was considered verified if the results matched to at least three decimal places of the results generated using SAS®.

The results of verifying estimates of model parameters, BMD, BMDL, and BMDU for each of the defined test cases are presented in tabular form within Sections 4.1 to 4.8, with each section addressing a particular model.[5] One row of each of these tables corresponds to a specific test case, whose data are specified within the first column in the input *,(d) file format, while the specific features targeted by the test cases are described in the second column. The format of the *.(d) input file for each model is explained in Appendix B. The results of comparing the estimated covariance/correlation matrices are presented in Appendix A.

4.1   Multistage Model with Background Dose Parameter

Initial Parameter Estimates: Initial estimates for the multistage model parameters were calculated by linear regression on a re-parameterized dose-response function. However, the addition of the background dose parameter to the model does not allow the dose-response function to be re-parameterized into a linear form in which a general algorithmic form can be readily coded. Therefore, the background dose parameter was set to a predetermined value of 0.5 to allow for linear regression to be used to estimate the remaining model parameters. Although an estimate of zero for the background dose parameter (as is used for other models) initially seemed reasonable, the functional form of the multistage and Cancer models caused the *donlp3* optimizer to get trapped within

---

[5] The cases tabulated represent the formal test cases. Note that other cases, not shown here, were examined informally for various reasons, e.g. to explore datasets that might cause convergence problems.

an artificial minimum that was imposed by use of either form of the safe log function (Section 3.7) in estimating the log-likelihood value. While more tests may be needed to find the optimal value for the initial estimate of background dose parameter, the current set of dose-response datasets resulted in no convergence issues when an initial value of 0.5 was established for the background dose parameter.

Scaling Parameters: As noted in Section 2.1, the beta parameters (coefficients of dose) were scaled by the maximum dose to minimize potential convergence-related problems.

Issues: The multistage model was found to be very stable when fitted to the test cases specified in Table 2. That is, while the *donlp3* optimizer failed to converge for some datasets, a lack of convergence was observed among the test cases only when positive response at control dose exceeded 60%, which is considered atypical. For datasets having a large positive response at zero dose and requiring a convex (sublinear) response curve, the original form of the model (with background response parameter) is expected to fit better than the alternative form (with background dose parameter). Thus, users should evaluate goodness-of-fit when choosing the appropriate model form for a given dataset. Model fit issues associated with high response at zero control dose are discussed further in Battelle (2007).

Table 2. Summary of Verification Tests Performed on the Multistage Model with Background Dose

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
MULTISTAGE01_14.(d) ------------------------------ 5 3 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 3 7 0.5 4 6 0.8 7 3 1.0 9 1	MLE calculation of background dose and three Beta parameters. Correlation Matrix, BMD, BMDL and BMDU calculations for Extra Risk type. Beta's restricted to greater than zero.	BG= 0.4408 Beta1=0.320 Beta2 =0.0 Beta3=0.468 BMD= 0.1510 BMDL= 0.06127 BMDU = 0.207	BG =0.440805 Beta1 = 0.320712 Beta2 = 0.0 Beta3 = 0.468104 BMD= 0.151004 BMDL= 0.0612734 BMDU = 0.20703	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
MULTISTAGE01_15.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 4 6 0.5 4 6 0.8 7 3 1.0 9 1	Three parameter model with non-increasing/flat dose response data.	BG= 0.544 Beta1= 0.00 Beta2=0.721 BMD= 0.12078 BMDL= 0.05423 BMDU =0.1834	BG = 0.5443 Beta1 = 1.92333e-008 Beta2 = 0.721278 BMD= 0.120784 BMDL= 0.0542246 BMDU = 0.183636	PASS
MULTISTAGE01_16.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.5 4 6 1.0 5 5 2.0 6 4 4.0 9 1	Test parameter scaling with Dose Max greater than 1.	BG= 0.738 Beta1= 0.320 Beta2= 0.028 BMD=0.2847 BMDL=0.14562 BMDU = 0.6	BG = 0.738255 Beta1 = 0.320422 Beta2 = 0.0281917 BMD= 0.284701 BMDL= 0.145626 BMDU = 0.626873	PASS
MULTISTAGE01_17.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 1.0 2 8 2.0 2 8 3.0 6 4 4.0 9 1	Test fitting non-increasing/flat dose response to background dose model. The dose-response data is flat for initial dose levels with a sudden increase in response at higher dose levels.	BG= 1.60 Beta1=0.00 Beta2= 0.043 BMD= 0.6272 BMDL=0.35125 BMDU=0.89	BG = 1.6001 Beta1 = 1.7073e-021 Beta2 = 0.0438912 BMD= 0.627189 BMDL= 0.351252 BMDU =0.89448	PASS
MULTISTAGE01_18.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.01000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 1.0 3 7 2.0 4 6 3.0 6 4 4.0 9 1	Fit data to a three parameter model with Beta's restricted to greater than zero. Test that the Beta's are evaluated at the boundary.	BG= 1.890 Beta1= 0.00 Beta2= 0.045 BMD = 0.05736 BMDL= 0.02459 BMDU = 0.084	BG = 1.89095 Beta1 = 0.0 Beta2 = 0.0456357 BMD = 0.0573626 BMDL= 0.0245904 BMDU = 0.08445	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
MULTISTAGE01_19.(d) Same data as 18.d	Same data as 18.d but test BMDL and BMDU calculation for Added Risk type.	BMD=0.067 BMDL= 0.05 BMDU = 0.108	BMD = 0.0674138 BMDL = 0.0503775 BMDU = 0.10829	PASS
MULTISTAGE01_20.(d) 5 3 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 0.44 -9999 -9999 -9999 0 -9999 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 3 7 0.5 4 6 0.8 7 3 1.0 9 1	Same data as MULTISTAGE01_14, but with background dose parameter fixed. Test that the fixed parameters are read correctly from input file and proper constraints are placed on the parameter during optimization for MLE and BMDL calculation	BG= 0.44 Beta1= 0.320 Beta2= 0.0 Beta3= 0.468	BG = 0.44 Beta1 = 0.320717 Beta2 = 0.0 Beta3 = 0.468108	PASS
MULTISTAGE01_21.(d) MULTISTAGE01 5 3 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 7 3 0.2 7 3 0.5 7 3 0.8 6 4 4.0 12 1 0.05	Test the model with greater than 60% response at control dose. This test case is used to illustrate that the model fails to converge for some atypical dose-response datasets.	Program execution is stopped cleanly with a failure message	Error Message – Program execution is stopped	PASS
MULTISTAGE01_22.(d) 5 3 250 1e-8 1e-8 1 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 1098 0.2 100 1900 0.5 700 1300 0.8 1300 700 1.0 1900 100 0.05	Test the model with large sample group (N = 2000)	BG= 0.0096 Beta1= 0.1663 Beta2= 0.0 Beta3= 2.2028	BG = 0.00961 Beta1 = 0.16632 Beta2 = 3.01128E-23 Beta3 = 2.20279	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
MULTISTAGE01_23.(d) MULTISTAGE01 5 3 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 6 4 0.2 6 4 0.5 7 3 0.8 5 5 1.0 4 6	Test the model with >50% positive response at control dose and a non-monotonic response curve. Convergence test, validation results not computed.	No convergence issues	BG = 8.77146 Beta1 = 0.088 Beta2 = 3.905E-18 Beta3 = 0	PASS
MULTISTAGE01_24.(d) MULTISTAGE01 5 3 250 1e-8 1e-8 1 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 1098 0.2 50 950 0.5 700 1300 0.8 130 70 1.0 190 10 0.05	Test the model with unequal sample groups at each dose levels. The number of test subjects varies from 100 - 2000	BG = 0.0624 Beta1 = 0.0 Beta2 = 0.3577 Beta3 = 1.7170	BG = 0.0623022 Beta1 = 1.32349E-22 Beta2 = 0.35765 Beta3 = 1.71702	PASS
MULTISTAGE01_25.(d) Same data as Multistage01_14.(d), but with BETA1 parameter specified as 0.33	Test the code with BETA1 parameter specified. This test case is to ensure that the code reads in the specified parameters in right order and scales them appropriately.	BG = 0.433 Beta1 = 0.33 (specified) Beta2 = 0 Beta3 = 0.4719	BG = 0.4332 Beta2 = 0 Beta3 = 0.472	PASS
MULTISTAGE01_26.(d) Same data as Multistage01_14.(d), but with BETA3 parameter specified as 0.47	Test the code with BETA3 parameter specified. This test case is to ensure that the code reads in the specified parameters in right order and scales them appropriately.	BG = 0.438 Beta1 = 0.33 Beta2 = 0 Beta3 = 0.47(specified)	BG = 0.4383 Beta1 = 0.323 Beta2 = 0 Beta3 = 0.47(specified)	PASS
MULTISTAGE01_27.(d) DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 100 50 15 85 100 30 70 150 65 35 200 100 0	Test the robustness of the model with dataset has zero response at control dose and 100% response at max dose	BG = 0.0 Beta1 = 0.002 Beta2 = 0.0 Beta3 = 0.0 Beta4 = 0.0	BG = 7.08245e-019 Beta1 = 0.00215809 Beta2 = 1.23746e-025 Beta3 = 0.0 Beta4 = 1.85845e-009	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
MultiStage01_28.(d) DOSE EFFECT1 NEGATIVE_RESPONSE 0 100 0 50 85 15 100 70 30 150 35 65 200 0 100	Test the model with monotonically decreasing dataset, 100% response at control dose and 0 response at max dose	BG = 932.0 Beta1 = 0.001 Beta2 = 0.0 Beta3 = 0.0 Beta4 = 0.0	BG = 932.418 Beta1 = 0.000853211 Beta2 = 0 Beta3 = 0 Beta4 = 2.7143e-026	PASS
Multistage01_1.(d) DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 100 50 5 95 100 30 70 150 65 35 200 90 10 0.05 150 0	Compare the Background Dose model with background response model, with the background parameters restricted to zero	BG = 0.0 Beta1 = 0.0 Beta2 = 4.43664e-6 Beta3 = 2.66519e-7 Beta4 = 0.0	BG = 0.0 Beta1 = 0.0 Beta2 = 4.43664e-6 Beta3 = 2.66519e-7 Beta4 = 0.0	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.1).

4.2   Cancer Model with Background Dose Parameter

The Cancer model is equivalent to the multistage model with the added restriction that the beta parameters must be greater than zero. Therefore, except for calculation of the Cancer slope factor and the BMDU, and for accounting for this restriction, the same types of changes made to the multistage model code were made to the Cancer model code when preparing a module corresponding to the Cancer model with background dose parameter. In addition, similar issues faced while testing the multistage model were encountered when fitting the Cancer model. The algorithm used to calculate the BMDU is similar to that used to calculate the BMDL, and no new issues were encountered in calculating the BMDU for the Cancer model.

Test cases used in verifying estimates generated by fitting the Cancer model with background dose parameter are listed in Table 3.

Table 3. Summary of Verification Tests Performed on the Cancer Model with Background Dose

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
CANCER01_01.(d) 5 3 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 -9999 0 -9999 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 3 7 0.5 4 6 0.8 7 3 1.0 9 1	Test the code when Beta's are not restricted in input (*.d) file.	Code should fail gracefully with a note about Beta's not restricted in the .out file.	Invalid Parameter entered in .d file . For Cancer Model, restrict should be 1 (Beta's should be > 0)	PASS
CANCER01_02.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 4 6 0.5 4 6 0.8 7 3 1.0 9 1	Test BMDU and Cancer Slope Factor calculation for a three parameter model.	BMDU= 0.183 Cancer Slope = 1.84433	BMDU = 0.183636 Cancer Slope = 1.84418	PASS
CANCER01_03.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.5 4 6 1.0 5 5 2.0 6 4 4.0 9 1	Test parameter scaling with Dose Max greater than 1.	BG= 0.738 Beta1=0.320 Beta2=0.028 BMD=0.284 BMDL=0.145 BMDU=0.629	BG = 0.738242 Beta1 = 0.320428 Beta2 = 0.0281904 BMD = 0.284699 BMDL = 0.145626 BMDU =0.626873	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
CANCER01_04.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 1.0 2 8 2.0 2 8 3.0 6 4 4.0 9 1	Test fitting non-strictly-monotonic dose response to background dose model. The dose-response data is flat for initial dose levels with a sudden increase in response at higher dose levels.	BG = 1.60 Beta1 = 0.0 Beta2 = 0.043 BMD = 0.627 BMDL= 0.3514 BMDU= 0.894 CSF = 0.28457	BG = 1.6001 Beta1 = 5.494e-011 Beta2 = 0.0438912 BMD = 0.627189 BMDL= 0.351252 BMDU= 0.89448 CSF = 0.284696	PASS
CANCER01_05.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.01000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 1.0 3 7 2.0 4 6 3.0 6 4 4.0 9 1	Test BMDL and BMDU calculation at a lower benchmark response level of 0.01.	BMDL= 0.024 BMDU=0.084	BMDL =0.0245906 BMDU =0.0844514	PASS
CANCER02_01.(d) 5 2 250 1e-8 1e-8 0 1 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 4 6 0.5 4 6 0.8 7 3 1.0 9 1	Same data as CANCER01_02.d but test BMDU calculation for Added Risk type.	BMDU= 0.265	BMDU = 0.265448	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.1).

4.3   Logistic Model with Background Response Parameter

In the current version of BMDS, a "logtrans" indicator is included within the input *,(d) file for the logistic model which determines whether a logistic or log-logistic model is fitted to the input data. In contrast, separate modules were prepared for the logistic model with background response parameter and log-logistic model with background dose parameter in order to ensure better code maintainability and to maintain consistency. Thus, the "logtrans" indicator is not used with the new module.

Initial Parameter Estimates: Initial estimates of the model parameters were calculated by linear regression on a re-parameterized objective function. A value of zero was specified as the initial estimate for the background response parameter.

Issues: No particular convergence issues were encountered within the specified test cases. The logistic with background response model resulted in better fits for dose-response data with large positive response at control dose than the existing logistic model (representing an implicit background dose model). There are issues with model fit at high response (50% or higher) at zero control dose. These are documented in Battelle (2007).

Scaling parameters: To limit the dose values in the objective function to less than 1 in order to improve convergence, the slope parameter is scaled by the maximum dose value.

Test cases for the logistic model with background response parameter are listed in Table 4.

Table 4. Summary of Verification Tests Performed on the Logistic Model with Background Response

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGISTIC01_19.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.5 2 8 1.0 2 8 2.0 6 4 4.0 9 1	Test MLE estimation for background response and other parameters for non-strictly-monotonic dose response data with positive response at zero dose.	BG = 0.1021 Intercept = -2.5977 Slope = 1.2305 BMD = 0.77767 BMDL = 0.409 BMDU = 1.9536	BG = 0.102125 Intercept = -2.59769 Slope = 1.23048 BMD = 0.777669 BMDL = 0.409144 BMDU = 1.9538	PASS
LOGISTIC01_20.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 **1** 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.5 2 8 1.0 2 8 2.0 6 4 4.0 9 1	Same data as 01_19.d file, but test BMD, BMDL and BMDU calculation for Added Risk type	BMD = 0.88243 BMDL = 0.488 BMDU = 2.128	BMD = 0.882433 BMDL = 0.488555 BMDU = 2.12801	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGISTIC01_21.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 3 7 0.5 4 6 0.8 7 3 1.0 9 1	Test model MLE fit with monotonic data and no scaling with max dose set to 1. (This was an early test; estimates of BMDU and BMDL were later considered in test case LOGISTIC01_22.(d).)	BG = 0.2216 Intercept = -4.175 Slope = 5.941	BG = 0.221582 Intercept = -4.17499 Slope = 5.94097	PASS
LOGISTIC01_22.(d) Same data as 01_21.(d)	Same data as 01_21.(d) but test BMDL, BMDU for Added Risk	BMDL = 0.1460 BMDU = 0.765	BMDL = 0.146117 BMDU = 0.76468	PASS
LOGISTIC01_25.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 4 6 0.5 4 6 0.8 7 3 1.0 9 1	Test MLE fit, BMDL and BMDU calculation for non-strictly-monotonic data with very flat response curve for initial dose level	BG = 0.2727 Intercept = -4.490 Slope = 6.1766 BMD = 0.388 BMDL = 0.109 BMDU = 0.7671	BG = 0.272605 Intercept = -4.48994 Slope = 6.17659 BMD = 0.388412 BMDL = 0.109271 BMDU = 0.769056	PASS
LOGISTIC01_27.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 4 6 0.5 3 7 1.0 5 5 2.0 7 3 4.0 9 1	Test the model, with <50% positive response at control dose.	No convergence issues are expected	BG = 0.1827 Intercept = -1.4329 Slope = 0.8876	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGISTIC01_28.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 998 0.5 97 903 1.0 350 650 2.0 670 330 4.0 900 100	Test the model with large sample group (N=1000)	BG = 0 Intercept = -2.5003 Slope = 1.4228	BG = 0.0 Intercept = -2.50027 Slope = 1.4227	PASS
LOGISTIC01_29.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 998 0.5 97 903 1.0 175 325 2.0 670 330 4.0 90 10	Test the model with unequal sample groups at each dose levels. The number of test subjects varies from 100 - 1000	BG = 0.0 Intercept = -3.1905 Slope = 1.975	BG = 0.0 Intercept = -3.19047 Slope = 1.9754	PASS
LOGISTIC01_30.(d) Same data as Logistic_19.(d), but with Background Response parameter specified at 0.1	Test the code with background response parameter specified, to ensure that the specified parameters are properly read and scaled within the code.	BG = 0.1 (Specified) Intercept = -2.58 Slope = 1.22	BG = 0.1 (specified) Intercept = -2.578 Slope = 1.2250	PASS
LOGISTIC01_31.(d) Same data as Logistic_19.(d), but with Intercept parameter specified at -2.6	Test the code with Intercept parameter specified, to ensure that the specified parameters are properly read and scaled within the code.	BG = 0.10 Intercept = -2.6 (specified) Slope = 1.23	BG = 0.1023 Intercept = -2.6 Slope = 1.231	PASS
LOGISTIC01_32.(d) Same data as Logistic_19.(d), but with Slope parameter specified at 1.2	Test the code with Slope parameter specified, to ensure that the specified parameters are properly read and scaled within the code.	BG = 0.10 Intercept = -2.51 Slope = 1.2 (specified)	BG = 0.095 Intercept = -2.514 Slope = 1.2 (specified)	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGISTIC01_33.(d) DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 10 0.5 2 8 1.0 4 6 2.0 6 4 4.0 10 0	Test the robustness of the code by fitting dataset with zero response at control dose and 100% response at max dose. The model cannot be used to fit monotonically decreasing datasets as the slope is restricted to be positive.	BG = 0.0 Intercept = -2.552 Slope = 1.685	BG = 0.0 Intercept = -2.5524 Slope = 1.68499	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.2).

4.4 Log-Logistic Model with Background Dose Parameter

As noted in Section 4.3, the current version of BMDS uses a "logtrans" indicator included within the input *,(d) file for the logistic model which determines whether a logistic or log-logistic model is fitted to the input data. Because a separate module was prepared for the log-logistic model with background dose parameter, the "logtrans" indicator is not used with this new module.

Initial Parameter Estimates: Initial estimates of the model parameters were calculated by linear regression on a re-parameterized objective function. A value of zero was specified as the initial estimate for the background dose parameter.

Issues: Several issues were encountered while attempting to validate results produced by the newly developed BMDS module and SAS®. The module frequently failed to converge for data having a large positive response at zero dose (control) or had model parameters that are very large and unrealistic (for example, when the background dose parameter was an order of magnitude larger than the maximum dose). These issues appear to be related to model fit problems and are discussed at length in Section 2 of Battelle (2007). Because these model fit problems have not been fully resolved, EPA chose not to release this module.

Test cases for the logistic model with background response parameter are listed in Table 5.

Table 5. Summary of Verification Tests Performed on the Log-Logistic Model with Background Dose

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGLOGISTIC01_19.(d) LOGLOGISTIC1 5 250 1e-8 1e-8 0 1 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.5 2 8 1.0 2 8 2.0 6 4 4.0 9 1	Test MLE estimation for background dose and other parameters for non-strictly-monotonic dose response data with positive response at zero dose.	BG = 1 Intercept = -2.230 Slope = 2.375	BG = 1 Intercept = -2.23 Slope = 2.37561	PASS

LOGLOGISTIC01_20.(d)
 	
5
250 1e-8 1e-8 0 1 0 1 0 0
0.1000 0 0.950
-9999 -9999 -9999
0
-9999 -9999 -9999
DOSE EFFECT1
NEGATIVE_RESPONSE
0   0 10
0.5 2 8
1.0 4 6
2.0 6 4
4.0 9 1
	Test MLE estimation for data with zero positive response at zero dose.	BG = 0.003
		Intercept =
		   -0.4216
		Slope= 1.597
			Background = 0.003136
			Intercept =
			   -0.421534
			Slope = 1.59713
				PASS

LOGLOGISTIC01_21.(d) Same data as 01_20.(d)	Same data as 01_20.(d), test BMDL and BMDU calculation for Added Risk type.	BMDL = 0.066 BMDU = 0.6381	BMD = 0.32598 BMDL = 0.0662 BMDU = 0.6381	PASS
LOGLOGISTIC01_23.(d) LOGLOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 3 7 0.5 4 6 0.8 7 3 1.0 9 1	Test scaling dose values with max dose set at 1.0	BG = 1.0 Intercept = -1.731 Slope = 4.570	BG = 1 Intercept = -1.7316 Slope = 4.57013	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGLOGISTIC01_24.(d) LOGISTIC1 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 3 7 0.5 4 6 0.8 7 3 1.0 9 1	Same data as 01_23.(d), test BMDL and BMDU calculations for Added Risk type.	BMD = 0.1491 BMDL = 0.052 BMDU = 0.2107	BMD = 0.149069 BMDL = 0.0524 BMDU = 0.21077	PASS
LOGLOGISTIC01_28.(d) LOGLOGISTIC1 5 250 1e-8 1e-8 0 1 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 3 997 0.1 30 970 0.5 90 910 3.0 460 540 4.0 900 100	Test the model with large sample group (N = 1000)	BG = 1 Intercept = -4.1789 Slope = 3.3595	BG = 1 Intercept = -4.1788 Slope = 3.35949	PASS
LOGLOGISTIC01_29.(d) LOGLOGISTIC1 5 250 1e-8 1e-8 0 1 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 50 50 0.5 50 50 1.0 60 40 2.0 70 30 4.0 90 10	Test the background dose model with 50% positive response at control dose. Objective of this test is to show that the optimizer is able converge for MLE estimates. Convergence test, validation results not computed.	No convergence issues	BG = 1 Intercept = -0.2907 Slope = 1.22508 Goodness of Fit Chi^2 = 7.19 (No convergence issues)	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGLOGISTIC01_30.(d) LOGLOGISTIC1 5 250 1e-8 1e-8 0 1 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 3 997 0.1 30 970 0.5 45 455 3.0 46 54 4.0 90 10 1	Test the model with unequal sample groups at each dose level.	BG = 0.142 Intercept = -1.442 Slope = 1.7393	BG = 0.142007 Intercept = -1.4412 Slope = 1.73931	PASS
LOGLOGISTIC01_31.(d) LOGLOGISTIC1 5 250 1e-8 1e-8 0 1 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 6 4 0.5 7 3 1.0 6 4 2.0 7 3 4.0 9 1	Test the model with >50% response at control dose. This test case is to illustrate that the model fails to converge for certain instances of atypical dose-response data. Convergence test, validation results not computed.	Program fails cleanly with a error message	Error Message – Program execution is stopped	PASS
LOGLOGISTIC01_32.(d) Same data as LogLogistic01_20.(d), with Background dose parameter fixed at 0.1	Test the model with specified parameters to ensure the specified values are read and scale appropriately	BG = 0.1 (specified) Intercept = -0.647 Slope = 1.80	BG = 0.1 Intercept = -0.6568 Slope = 1.803	PASS
LOGLOGISTIC01_33.(d) LOGLOGISTIC1 5 250 1e-8 1e-8 0 1 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 10 0.5 2 8 1.0 4 6 2.0 6 4 4.0 10 0	Test the robustness of the code by fitting dataset with zero response at control dose and 100% response at max dose. The model is not suitable for fitting monotonically decreasing datasets.	BG = 0.247 Intercept = -1.03 Slope = 2.49	BG = 0.246977 Intercept = -1.02917 Slope = 2.4977	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.2).

4.5 Probit Model with Background Response Parameter

Like the logistic and log-logistic models, the current version of BMDS includes a "logtrans" indicator within the input *,(d) file for the probit model which determines whether a probit or log-probit model is fitted to the input data. In contrast, separate modules were prepared for the probit model with background response parameter and log-probit model with background dose parameter in order to ensure better code maintainability and to maintain consistency. Thus, the "logtrans" indicator is not used with the new module.

Initial Parameter Estimates: Initial estimates of the model parameters were calculated by linear regression on a re-parameterized objective function. A value of zero was specified as the initial estimate for the background response parameter.

Issues: No particular convergence issues were encountered while testing. Some model fit issues occurred when high response rates (50% or higher) occurred at the control dose, which are demonstrated in Battelle (2007).

Test cases for the probit model with background response parameter are listed in Table 6.

Table 6. Summary of Verification Tests Performed on the Probit Model with Background Response

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
PROBIT_1.(d) PROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.5 2 8 1.0 2 8 2.0 6 4 4.0 9 1	Test MLE estimation for background dose and other parameters for non-strictly-monotonic dose response data with positive response at zero dose.	BG = 0.1192 Intercept = -1.6139 Slope = 0.7350 BMD = 0.77357 BMDL = 0.402 BMDU = 1.995	BG = 0.119198 Intercept = -1.61388 Slope = 0.73505 BMD = 0.773569 BMDL =0.402653 BMDU = 1.99675	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
PROBIT_2.(d) PROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 8 0.2 4 6 0.5 4 6 0.8 7 3 1.0 9 1	Test BMDL, BMDU calculations for Added Risk type.	BMDL = 0.1369 BMDU = 0.804	BMDL =0.136899 BMDU = 0.80468	PASS
PROBIT_7.(d) PROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 200 800 0.2 250 750 1.0 200 800 2.0 630 370 4.0 990 10	Test the model with large sample group (N = 1000)	BG = 0.2010 Intercept = -2.6908 Slope = 1.3222	BG = 0.20101 Intercept = -2.69079 Slope = 1.32216	PASS
PROBIT_8.(d) PROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 200 800 0.2 250 750 1.0 100 400 2.0 63 37 4.0 99 1	Test the model with unequal number of sample group at each dose level	BG = 0.2164 Intercept = -3.4696 Slope = 1.6446	BG = 0.2163 Intercept = -3.4695 Slope = 1.6446	PASS
PROBIT_9.(d) Same data as PROBIT_1.(d), with background response parameter specified as 0.12	Test model with specified parameters, to ensure the parameters are read in the correct order and scaled appropriately	BG = 0.12(Specified) Intercept = -1.62 Slope = 0.736	BG = 0.12 Intercept = -1.618 Slope = 0.7362	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
PROBIT_10.(d) Same data as PROBIT_1.(d), with Intercept parameter specified as -1.6	Test model with specified parameters, to ensure the parameters are read in the correct order and scaled appropriately	BG = 0.118 Intercept = -1.6 (specified) Slope = 0.731	BG = 0.11718 Intercept = -1.6 Slope = 0.7310	PASS
PROBIT_11.(d) Same data as PROBIT_1.(d), with Slope parameter specified as 0.73	Test model with specified parameters, to ensure the parameters are read in the correct order and scaled appropriately	BG = 0.118 Intercept = -1.60 Slope = 0.73 (Specified)	BG = 0.1171 Intercept = -1.598 Slope = 0.73	PASS
PROBIT_12.(d) DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 10 0.5 2 8 1.0 2 8 2.0 6 4 4.0 10 0	Test the robustness of the code by fitting dataset with zero response at control dose and 100% response at max dose. The model is unable to fit datasets with decreasing dose-response curve.	BG = 0.0 Intercept = -1.777 Slope = 1.058	BG = 0.0 Intercept = -1.77696 Slope = 1.05853	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.3).

4.6   Log-Probit Model with Background Dose Parameter

As noted in Section 4.5, the current version of BMDS uses a "logtrans" indicator included within the input *,(d) file for the probit model which determines whether a probit or log-probit model is fitted to the input data. Because a separate module was prepared for the log-logistic model with background dose parameter, the "logtrans" indicator is not used with this new module.

Initial Parameter Estimates: The initial estimate of the background dose parameter was set to zero. This allowed for use of existing algorithms in BMDS to estimate the initial values of other parameters. No convergence issues were encountered while testing as a result of setting the initial estimate of background dose to zero.

Issues: Convergence issues were seen in some test cases with the log-probit model when fitting to data with large response (>50%) at control dose. Issues affecting convergence and model fit are explored in Battelle (2007).

Test cases for the log-probit model with background dose parameter are listed in Table 7. The problem with convergence is noted for one test case in this table.

Table 7.   Summary of Verification Tests Performed on the Log-Probit Model with Background Dose

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGPROBIT_1.(d) LOGPROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 10 0.5 2 8 1.0 2 8 2.0 6 4 4.0 9 1	Test MLE estimation for background dose and other parameters for non-strictly-monotonic dose response data.	BG = 0.5298 Intercept = -1.236 Slope = 1.6129 BMD = 0.482 BMDL = 0.1696 BMDU = 0.893	BG = 0.529767 Intercept = -1.23675 Slope =1.61284 BMD = 0.482766 BMDL = 0.169529 BMDU = 0.89335	PASS
LOGPROBIT_2.(d) LOGPROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 10 0.5 2 8 1.0 2 8 2.0 6 4 4.0 9 1	Same data as LOGPROBIT_1.d, except test BMDL calculation for Extra Risk type.	BMDL = 0.1693 BMDU = 0.8780	BMDL = 0.169283 BMDU = 0.878091	PASS
LOGPROBIT_3.(d) 5 250 1e-8 1e-8 0 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 10 0.5 4 6 1.0 5 5 2.0 6 4 4.0 9 1	Test parameter estimation with Slope parameter restricted to greater than 1	BG = 0.1266 Intercept = -0.151 Slope = 1	BG = 0.126555 Intercept = -0.1511 Slope = 1	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGPROBIT_4.(d) 5 250 1e-8 1e-8 0 1 1 1 1 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 dose Y4 NEGATIVE_RESPONSE 0 0 10 1 3 7 2 6 4 3 9 1 4 4 6	Test parameter Estimation with Slope restricted to > 1	BG= 0.1722 Intercept= -0.764 Slope=1.0	BG=0.172243 Intercept = -0.7644 Slope = 1	PASS
LOGPROBIT_5.(d) 5 250 1e-8 1e-8 0 1 1 1 1 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 dose Y4 NEGATIVE_RESPONSE 0 1 9 1 3 7 2 6 4 3 9 1 4 4 6	Test parameter estimation with Slope > 1 and positive response at zero dose	BG = 0.836 Intercept = -1.029 Slope = 1	BG = 0.83656 Intercept = -1.029 Slope = 1	PASS
LOGPROBIT_6.(d) 5 250 1e-8 1e-8 0 1 1 1 1 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 dose Y8 NEGATIVE_RESPONSE 0 2 8 1 5 5 2 8 2 3 10 0 4 4 6	Test the background dose model with positive response (<50%) at control dose and a non-monotonic dose-response curve. Convergence test, validation results not computed.	No convergence issues are expected	BG = 1.2269 Intercept = -0.8498 Slope = 1 (No convergence issues)	PASS
LOGPROBIT_7.(d) Y8-LPR 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 dose Y8 NEGATIVE_RESPONSE 0 6 4 0.1 7 3 0.2 8 2 0.8 9 1 4.0 7 3	Test the model with >50% response at control dose. This dose response data is an illustration of the atypical data that might cause converge issues. Convergence test, validation results not computed.	Program exits cleanly with an error message	Error Message – Fitted Parameters failed to converge	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
LOGPROBIT_8.(d) LOGPROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 1000 0.2 200 800 1.0 370 630 2.0 600 400 4.0 990 100	Test the model with large sample group (N = 1000)	BG = 18.8691 Intercept = -54.2679 Slope = 18	BG = 18.8697 Intercept = -54.2697 Slope = 18	PASS
LOGPROBIT_9.(d) LOGPROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 1000 0.2 200 800 1.0 370 630 2.0 300 200 4.0 80 20	Test the model with unequal number of sample groups at different dose levels.	BG = 0.0 Intercept = -0.1819 Slope = 0.4540	BG = 0.0 Intercept = -0.181804 Slope = 0.454016	PASS
LOGPROBIT_10.(d) Same data as LOGPROBIT_1.(d), but with Background Dose specified at 0.5	Test the model with specified parameters to ensure the parameters are read in the right order and scale appropriately.	BG = 0.5 (specified) Intercept = -1.19 Slope = 1.58	BG = 0.5 Intercept = -1.188 Slope = 1.583	PASS
LOGPROBIT_11.(d) Same data as LOGPROBIT_1.(d), but with Intercept parameter specified at -1.2	Test the model with specified parameters to ensure the parameters are read in the right order and scale appropriately.	BG = 0.50 Intercept = -1.2 (Specified) Slope = 1.59	BG = 0.507 Intercept = -1.2 Slope = 1.5904	PASS
LOGPROBIT_12.(d) Same data as LOGPROBIT_1.(d), but with Slope parameter specified at 1.6	Test the model with specified parameters to ensure the parameters are read in the right order and scale appropriately.	BG = 0.52 Intercept = -1.21 Slope = 1.6 (Specified)	BG = 0.5173 Intercept = -1.216 Slope = 1.6	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
<u>LOGPROBIT_13.(d)</u> LOGPROBIT 5 250 1e-8 1e-8 0 0 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 10 0.5 2 8 1.0 3 7 2.0 6 4 4.0 10 0	Test the robustness of the code by fitting dataset with zero response at control dose and 100% response at max dose.	BG = 15.95 Intercept = -51.55 Slope = 18	BG = 15.9518 Intercept = -51.5485 Slope = 18	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.3).

4.7   Weibull Model with Background Dose Parameter

Issues: The module for fitting the Weibull model with background dose parameter was tested by considering several test datasets with large positive responses at control dose. In each case, the model was successful in converging to a minimum. However, when the response rate at the control dose was small but non-zero (2% or less in our limited testing) or was high (50% or more in limited testing), the model fit parameters were imprecise. This is due to the MLE occurring in a flat region of the optimization surface and is discussed in greater detail within Battelle (2007). In these situations, BMDS gives the following warning:

```
"WARNING: MLE Estimates are imprecise, donlp2 message = stepsizeselection: no
acceptable stepsize in [sigsm,sigla]"
```

When imprecise Weibull model fits are encountered, the user should review goodness of fit statistics and make a qualitative judgment on the validity of the model to fit the particular dataset.

BMDU estimation: The *donlp3* optimizer routinely failed to converge while performing a constrained optimization (i.e., maximization) to find the BMDU at a specified confidence level. To overcome this limitation, an alternative approach was used which estimated BMDU to within a range based on the constraint on the Log Likelihood at the given confidence level. This approach assumed that the Log Likelihood function was monotonic at least within the range of the BMD. The best estimate of BMDU was found within that range using the *zeroin* function (a root finding function in the BMDS utility library). If the MLE constraint was not satisfied within the precision of 1E-8, then the algorithm was repeated within a 10% range of the latest BMDU estimate. If the BMDU estimate failed to converge, then it was reported with the following warning message:

Warning: BMDU computation is at best imprecise for these data, Error = -0.005896

Validation of BMDL and BMDU Estimates: Third party statistical tools like SAS® or WinBUGS (which was explored for this case) either routinely failed or were unable to estimate BMDU and BMDL. SAS® was able to provide an estimate for BMDL and BMDU only for few data sets and failed to converge for several other cases that were tested. Reasons for convergence failures include those addressed in Battelle (2007). The results from BMDS were validated with SAS® for the datasets included in Table 9. It was also observed that changing the optimization routine used by SAS® would sometimes lead to convergence. However, changing routines on examples that did converge usually led to a change in the SAS® result. Thus, for most datasets tested, SAS® either failed to converge or gave inconsistent results depending on the type of optimization routine used (i.e., quasi-Newton optimization led to results which matched BMDS, while Nelder-Mead simplex optimization was more likely to converge, but resulted in poorer estimates than quasi-Newton when both converged successfully). Thus, even though the validation results in Table 9 for BMDU include examples where BMDS and SAS® agreed in their estimates, concern of the integrity of the SAS® results remains. (Note that Mathematica® was not considered here as the same algorithm would need to have been implemented within Mathematica®, and therefore, the results generated could not be used to conclude that the algorithm/approach used to estimate BMDU/BMDL was correct.)

Initial Parameter Estimates: In determining MLEs for the Weibull model with background dose, the following method was used to find starting values for the model parameters:

By transforming the Weibull probability model given in Table 1 and using a first order Taylor series approximation, the background dose Weibull model for dose d_i was linearized in terms of unknown model parameters as follows:

$$
\begin{aligned}
\ln[-\ln[1-\pi(d_i \mid \alpha,\beta,\eta)]] &= \ln[-\ln[1-(1-\exp\{-\alpha(d_i+\eta)^\beta\})]] \\
&= \ln[-\ln[\exp\{-\alpha(d_i+\eta)^\beta\}]] \\
&= \ln[\alpha(d_i+\eta)^\beta] \\
&= \ln\alpha + \beta\ln[d_i+\eta] \\
&\approx \ln\alpha + \beta(\ln d_i + \eta/d_i) \\
&= a + b\ln d_i + c/d_i
\end{aligned}
$$

where $a = \ln \alpha$, $b = \beta$, and $c = \beta\eta$. Define the "responses" for dosage level i as

$$
\begin{aligned}
y^i &= \ln[-\ln[1-\hat{\pi}^i]], \quad i=1,2,...,I \\
y_i &= \ln[-\ln[1-\hat{\pi}_i]], \quad i=1,2,...,I
\end{aligned}
$$

Estimates of a, b, and c, in the above equation were obtained by using least squares techniques to fit the following linear regression model:

$$
\begin{aligned}
y^i &= a + b\ln d^i + c(1/d^i), \quad i=1,2,...,I \\
y_i &= a + b\ln d_i + c(1/d_i), \quad i=1,2,...,I
\end{aligned}
$$

Consequently, if $\hat{a}$, $\hat{b}$, and $\hat{c}$, are least squares estimates for the regression model, then the initial estimates (i.e., starting values) for the slope, power, and background dose, parameters are $\hat{\alpha} = \exp\hat{a}$, $\hat{\beta} = \hat{b}$, and $\hat{\eta} = \hat{c}/\hat{b}$, respectively.

Depending on specified model restrictions and the data, some minor adjustments required by this method include:

- If $\hat{\pi}_i = 0$, then set the empirical estimate of the response probability to a very small value, e.g., a fraction of $1 / N_i$. If $\hat{\pi}_i = 1$, then set the probability to a value that is very close to 1, e.g., $1 - $ (a fraction of $1 / N_i$).
- If one of the dose levels is 0, then set the dose to a very small value, e.g., a fraction of the smallest, non-zero dose encountered in the data.
- If $\hat{\beta} < 0$, then set the starting value for the power parameter to 0. If, in addition, the model is specified with the restriction that $\beta \geq 1$, but $\hat{\beta} < 1$, then set the starting value for the power parameter to 1. If $\hat{\eta} < 0$, then set the starting value for the background dose parameter to 0.

Scaling: Parameter scaling did not appear to help alleviate any of the encountered convergence issues.

Test cases for the Weibull model with background dose parameter are listed in Table 8.

Table 8.   Summary of Verification Tests Performed on the Weibull Model with Background Dose

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
WEIBULL01.(d) WEIBULL01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 1 99 0.5 3 97 1.0 5 95 2.0 11 89 4.0 30 70	Test MLE estimation for background dose and other parameters for a monotonic dose response data.	BG = 0.9399 Slope = 0.0129 Power = 2.0769 BMD = 1.95249 BMDL = 1.41095	BG = 0.939874 Slope = 0.0128504 Power = 2.07695 BMD = 1.95249 BMDL = 1.41095	PASS
WEIBULL02.(d) WEIBULL01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 1 99 0.5 3 97 1.0 5 95 2.0 11 89 4.0 30 70	Same data as WEIBULL01.d, except test BMDL calculation for Added Risk type.	BMDL = 1.4180	BMDL = 1.41807	PASS
WEIBULL03.(d) 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 98 0.5 13 87 1.0 18 82 2.0 27 73 4.0 38 62	Test BMDU estimation for Extra Risk type	BMD = 0.4389 BMDU = 0.8025	BMD =0.438968 BMDU = 0.802555	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
WEIBULL04.(d) 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 100 0.5 3 97 1.0 5 95 2.0 8 92 4.0 14 86	Test BMDU estimation for Extra Risk type	BMD = 2.566076 BMDU = 4.81179	BMD = 2.56608 BMDU = 4.812	PASS
WEIBULL05.(d) 5 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 5 95 0.5 8 92 1.0 10 90 2.0 13 87 4.0 16 84	Test BMDU estimation	BMD = 3.28307 BMDU = 8.57184	BMD = 3.28307 BMDU = 8.57174	PASS
WEIBULL06.(d) WEIBULL01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 40 60 0.5 60 40 1.0 55 55 2.0 80 20 4.0 91 9	Test model with large positive response at control dose. Convergence test, validation results not computed.	No convergence issues.	BG = 2.41031 Slope = 0.13533 Power = 1.56706 (No convergence issues.)	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
WEIBULL08.(d) WEIBULL01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 1 999 0.2 30 970 2.5 250 750 3.0 410 390 4.0 600 400	Test the model with large sample group (N = 1000)	BG = 0.6419 Slope = 0.026 Power = 2.3635	BG = 0.64187 Slope = 0.02541 Power = 2.3634	PASS
WEIBULL09.(d) WEIBULL01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 1 999 0.2 30 970 2.5 125 375 3.0 410 390 4.0 60 40	Test the model with unequal number of sample groups at different dose levels	BG = 0.02833 Slope = 0.1476 Power = 1.2556	BG = 0.02833 Slope = 0.14759 Power = 1.25564	PASS
WEIBULL10.(d) Same data as WEIBULL01.(d), but Background Dose parameter specified at 0.9	Test the model with specified parameters to ensure the code reads in specified parameters in the correct order.	BG = 0.9 (Specified) Slope = 0.014 Power = 2.05	BG = 0.9 Slope = 0.0137 Power = 2.044	PASS
WEIBULL11.(d) Same data as WEIBULL01.(d), but Power parameter specified at 2.0	Test the model with specified parameters to ensure the code reads in specified parameters in the correct order.	BG = 0.85 Slope = 0.015 Power = 2.0 (Specified)	BG = 0.852 Slope = 0.01498 Power = 2.0	PASS
WEIBULL12.(d) DOSE EFFECT1 NEGATIVE_RESPONSE 0 0 100 0.5 30 90 1.0 50 50 2.0 71 29 4.0 100 0	Test the robustness of the code by fitting dataset with zero response at control dose and 100% response at max dose.	BG = 0.0 Slope = 0.638 Power = 1.28	BG = 0.0 Slope = 0.63842 Power = 1.27985	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
WEIBULL13.(d) DOSE EFFECT1 NEGATIVE_RESPONSE 0 1 99 1 50 50 5.0 90 10	Test the code changes made to the model to report approximate MLE values with a warning message with convergence precision of 1E-8 is not satisfied.	BG = 0.0035 Slope = 0.6914 Power = 0.7472	BG = 0.001 Slope = 0.6988 Power = 0.7418 `WARNING: MLE Estimates are imprecise, donlp2 message = stepsizeselection: no acceptable stepsize in [sigsm,sigla]`	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.4).

4.8   Gamma Model with Background Dose Parameter

Initial Parameter Estimates: In determining MLEs for the gamma model with background dose parameter, starting values for the model parameters were determined by the following algorithm. First, by transforming the gamma model probability, the following relationship holds:

$$
\begin{aligned}
G^{-1}(\pi(d_i \mid \alpha, \beta, \eta), \beta) &= G^{-1}(G(\alpha(d_i + \eta), \beta), \beta) \\
&= \alpha(d_i + \eta) \\
&= c + ad_i
\end{aligned}
$$

where $c = \alpha\eta$, $a = \alpha$, and $G^{-1}(\cdot, \tau)$ is the inverse cumulative distribution function of the standard gamma distribution with shape parameter τ. One of two possible approaches is then taken:

- If the distribution's power parameter β is assumed to be fixed, then a "response" at the ith dose level d_i is defined as

$$
y_i(\beta) = G^{-1}(\hat{\pi}_i, \beta)
$$

  Estimates of c and a are obtained by using least squares techniques to fit the following linear regression model:

$$
y_i(\beta) = c + ad_i
$$

  If $\hat{c}$ and $\hat{a}$ represents these least squares estimates, then initial estimates for the slope and background dose parameters are $\hat{\alpha} = \hat{a}$, and $\hat{\eta} = \hat{c}/\hat{a}$.

- If the value of β is not considered fixed, then an initial value of β is specified (e.g., 1.0 if $\beta > 0$, or 5.0 if $\beta > 1$), and the above regression is performed to

find $\hat{\alpha}$ and $\hat{\eta}$. Then, for each dose level, the following equations are solved for β_i

$$\hat{\pi}_i = G(\hat{\alpha}(d_i + \hat{\eta}), \beta_i) \quad i = 1, 2, ..., I$$

The mean of the β_i's is taken as the updated value for β. These regression steps are iterated until either all the parameters converge or the number of iterations reaches some pre-set limit.

Depending on specified model restrictions and the data, some minor adjustments may be required, such as the following:

- If $\hat{\pi}_i = 0$, then the empirical estimate of the response probability is set to a very small value (e.g., a fraction of $1 / N_i$). If $\hat{\pi}_i = 1$, then the probability is set to a value very close to 1 (e.g., $1 - $ (a fraction of $1 / N_i$)).
- If $\hat{\alpha} < 0$, then the starting value for the slope parameter is set to 0. If $\hat{\eta} < 0$, then the starting value for the background dose parameter is set to 0.

Issues:

The module used to calculate the covariance matrix for the gamma model with background dose parameter used numerical (finite difference) methods to calculate the differential of the objective function. Results generated using SAS® and results generated using the module differed by more than 10%. On further examination, it was found that errors in the numerical approximations that occur while calculating the second order differential of the objective function were further compounded by calculating the inverse of the Fisher Information Matrix to calculate the covariance matrix. To mitigate this problem, analytical methods were applied to differentiate the objective function, based on series expansion [Moore (1982)] or continued fraction expansion of the incomplete gamma integral.

The series expansion of the incomplete gamma integral has very good convergence [Pearson (1922)] when (dose + background dose) / alpha << 1, while the continued fraction expansion was found to have good convergence when (dose + background dose) / alpha >> 1. In cases where the ratio is comparable to one, both methods were found to be unsatisfactory, and finite difference approximations were used in such cases.

As with the other background dose quantal models there were model fit issues with high response at zero control dose. These issues are documented in Battelle (2007).

Test cases for the gamma model with background dose parameter are listed in Table 9.

Table 9. Summary of Verification Tests Performed on the Gamma Model with Background Dose

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
GAMMA01.(d) GAMMA01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 3 97 0.5 6 94 1.0 9 91 2.0 16 84 4.0 35 65	Test MLE estimation for background dose and other parameters for a monotonic dose response data.	BG = 1.7392 Slope = 0.3247 Power = 2.739 BMD = 1.5518 BMDL =1.00754 BMDU = 2.1296	BG = 1.73929 Slope = 0.324779 Power = 2.73979 BMD = 1.55158 BMDL = 1.00754 BMDU = 2.18718	PASS
GAMMA02.(d)	Same data as GAMMA01.d, except test BMDL and BMDU calculations for Added Risk type.	BMD = 1.59199 BMDL = 0.98626 BMDU = 2.114	BMD = 1.59199 BMDL =0.986259 BMDU = 2.11453	PASS
GAMMA03.(d) GAMMA01 5 250 1e-8 1e-8 0 1 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 2 98 0.5 7 93 1.0 11 89 2.0 16 84 4.0 24 76	Test MLE estimation when power is restricted to > 1	BG = 0.4127 Slope = 0.069 Power = 1.0 BMDL = 1.16 BMDU =2.39980	BG = 0.412739 Slope= 0.0690165 Power = 1 BMDL = 1.16013 BMDU =2.39979	PASS
GAMMA04.(d)	Same data as GAMMA03.d, except test BMDL calculation for Added Risk type.	BMDL = 1.1399 BMDU = 2.2880	BMDL = 1.13995 BMDU =2.2897	PASS
GAMMA05.(d) GAMMA01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 30 970 0.1 60 940 2.5 190 810 3.0 260 740 4.0 350 650	Test the model with large sample group (N = 1000)	BG = 7.2267 Slope = 0.9110 Power = 11.793	BG = 7.22738 Slope = 0.91108 Power = 11.795	PASS

Test Case Name and Definition[1]	Test Description	Expected Outcome	Actual Outcome	Test Pass/Fail
GAMMA06.(d) GAMMA01 5 250 1e-8 1e-8 0 0 1 0 0 0.1000 0 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 30 970 0.1 60 940 0.5 190 810 1.0 160 340 2.0 55 45	Test the model with unequal number of sample groups at different dose levels	BG = 0.1005 Slope = 0.4170 Power = 1.0905	BG = 0.1006 Slope = 0.41708 Power = 1.0904	PASS
GAMMA07.(d) GAMMA01 5 250 1e-8 1e-8 0 1 1 0 0 0.1000 1 0.950 -9999 -9999 -9999 0 -9999 -9999 -9999 DOSE EFFECT1 NEGATIVE_RESPONSE 0 58 42 0.5 30 70 1.0 61 39 2.0 82 18 4.0 91 9	Test the model with atypical dose-response data with large positive response (>50%) at control dose. Convergence test, validation results not computed.	No Convergence issues	BG = 96.45 Slope = 15.49 Power = 1501.11 Goodness-of-fit Chi^2 = 30.88 (No convergence issues.)	PASS
GAMMA08.(d) Same data as GAMMA01.(d) with Background Dose parameter specified at 1.74	Test the model with specified parameters to ensure that the code reads in the parameters in the correct order	BG = 1.74 (Specified) Slope = 0.325 Power = 2.74	BG = 1.74 Slope = 0.3248 Power = 2.740	PASS
GAMMA09.(d) Same data as GAMMA01.(d) with Power parameter specified at 2.8	Test the model with specified parameters to ensure that the code reads in the parameters in the correct order	BG = 1.787 Slope = 0.33 Power = 2.8 (Specified)	BG = 1.787 Slope = 0.3312 Power = 2.8	PASS

[1] Expressed as *.(d) input files in the format specified in Appendix B (Section B.5).

5.0   REFERENCES

1. Battelle (2007). Draft Convergence Report for Software Modules to Incorporate Background Dose/Response in Quantal Models within Benchmark Dose Modeling Software (BMDS). Prepared by D. Mooney for the National Center for Environmental Assessment, Office of Research and Development, US Environmental Protection Agency. WA 3-08, EPA Contract No. EP-C-04-027.

2. Moore, R. J. (1982). Algorithm AS 187. Derivatives of the Incomplete Gamma Integral *Appl. Statist*., Vol 31. 330 – 335.

3. Pearson, K. (1922). *Tables of the Incomplete Gamma Function*. London: Cambridge University Press.

4. Seber, G.A.F., and C.J. Wild. (1989). *Nonlinear Regression*. New York: Wiley-Interscience.

5. USEPA (2003). *Draft Guidance on the Development, Evaluation, and Application of Regulatory Environmental Models.* Prepared by The Council for Regulatory Environmental Modeling, Office of Science Policy, Office of Research and Development, U.S. Environmental Protection Agency, November 2003

6. Wheeler, M.W**.** (2005). Benchmark Dose Estimation Using SAS®. Paper No. 201-30, *Proceedings of the Thirtieth Annual SAS® Users Group International Conference*. Cary, NC: SAS Institute Inc. Available at http://www2.sas.com/proceedings/sugi30/201-30.pdf.

7. Wolfram Research, Inc., *Mathematica, Version 5.2*, Champaign, IL (2005).

Some of the output for this paper was generated using SAS/STAT software, Version 9.1.3 of the SAS System for Windows. Copyright 2002-2003, SAS Institute Inc. SAS and all other SAS Institute Inc. product or service names are registered trademarks or trademarks of SAS Institute Inc., Cary, NC, USA.

APPENDIX A

Numerical Results for Correlation Matrices Test Cases Generated Using SAS®

This appendix gives the data and numerical results of correlation matrix validation cases that were developed using SAS®. The values of other parameters used in validation found by SAS® or *Mathematica*® are reported in the tables in Section 4.

A.1   Multistage and Cancer Models with Background Dose

<u>Test Case 1</u> – MULTISTAGE_15.(d)

Expected Results:

```
Correlation Matrix of Parameter Estimates

Row    Parameter       _GAMMA      _BDOSE1      _BDOSE2

  1    _GAMMA          1.0000      -0.9730      -0.2162
  2    _BDOSE1        -0.9730       1.0000     0.000216
  3    _BDOSE2        -0.2162     0.000216      1.0000
```

Actual Results:

```
      Asymptotic Correlation Matrix of Parameter Estimates

                  Background_Dose        Beta(1)       Beta(2)

  Background_Dose              1          -0.97         -0.22

         Beta(1)          -0.97              1       1.8e-007

         Beta(2)          -0.22       1.8e-007              1
```

<u>Test Case 2</u> – MULTISTAGE_16.(d)

Expected Results:

```
            Correlation Matrix of Parameter Estimates

      Row    Parameter       _GAMMA      _BDOSE1      _BDOSE2
        1    _GAMMA          1.0000      -0.9607       0.8459
        2    _BDOSE1        -0.9607       1.0000      -0.9410
        3    _BDOSE2         0.8459      -0.9410       1.0000
```

Actual Results:

```
        Asymptotic Correlation Matrix of Parameter Estimates

                    Background_Dose        Beta(1)        Beta(2)

  Background_Dose               1          -0.96           0.85

         Beta(1)           -0.96              1          -0.94

         Beta(2)            0.85          -0.94              1
```

A.2   Logistic Model with Background Response

Test Case 1 – LOGISTIC_19.(d)

Expected Results:

```
Correlation Matrix of Parameter Estimates

   Row    Parameter        _ALPHA        _SLOPE        _BG_RESP

     1    _ALPHA           1.0000       -0.9222       -0.8735
     2    _SLOPE          -0.9222        1.0000        0.7366
     3    _BG_RESP        -0.8735        0.7366        1.0000
```

Actual Results:

```
Asymptotic Correlation Matrix of Parameter Estimates

            background_response      intercept        slope

background_response           1          -0.87          0.74
        intercept         -0.87              1          -0.92
            slope          0.74          -0.92              1
```

Test Case 2 – LOGISTIC_21.(d)

Expected Results:

```
Correlation Matrix of Parameter Estimates

            Row    Parameter        _ALPHA        _SLOPE         _BG_RESP

              1    _ALPHA           1.0000       -0.9826       -0.7639
              2    _SLOPE          -0.9826        1.0000        0.7046
              3    _BG_RESP        -0.7639        0.7046        1.0000
```

Actual Results:

```
Asymptotic Correlation Matrix of Parameter Estimates
```

```
              background_response     intercept          slope
background_response             1         -0.76            0.7
          intercept         -0.76            1           -0.98
              slope            0.7        -0.98              1
```

A.3   Log-logistic Model with Background Dose

Test Case 1 – LOGLOGISTIC_20.(d)

Expected Results:

```
Correlation Matrix of Parameter Estimates

            Row    Parameter      _ALPHA        _SLOPE        _BG_DOSE

             1     _ALPHA        1.0000       -0.5417       -0.5629
             2     _SLOPE       -0.5417        1.0000        0.4226
             3     _BG_DOSE     -0.5629        0.4226        1.0000
```

Actual Results:

```
Asymptotic Correlation Matrix of Parameter Estimates

             background_dose     intercept          slope
  background_dose            1         -0.58           0.43
        intercept        -0.58            1           -0.55
            slope          0.43        -0.55              1
```

A.4   Probit Model with Background Response

Test Case 1 – PROBIT_1.(d)

Expected Results:

```
          Correlation Matrix of Parameter Estimates

            Parameter    Inter       Slope        BG_RESP
              Inter       1.0000     -0.9303       -0.8887
              Slope      -0.9303      1.0000        0.7695
              BG_RESP    -0.8887      0.7695        1.0000
```

Actual Results:

```
Asymptotic Correlation Matrix of Parameter Estimates

              background_response     intercept          slope
background_response             1         -0.89           0.77
          intercept         -0.89            1           -0.93
              slope           0.77        -0.93              1
```

A.5   Log-Probit Model with Background Dose

Test Case 1 – LOGPROBIT_1.(d)

Expected Results:

```
          Correlation Matrix of Parameter Estimates

 Row     Parameter        Inter        Slope        BG_DOSE

  1      Inter           1.0000      -0.9862      -0.9919
  2      Slope          -0.9862       1.0000       0.9708
  3      BG_DOSE        -0.9919       0.9708       1.0000
```

Actual Results:

```
          Asymptotic Correlation Matrix of Parameter Estimates

                background_dose     intercept        slope
background_dose              1         -0.99         0.97
      intercept          -0.99             1        -0.99
          slope           0.97         -0.99            1
```

A.6   Weibull Model with Background Dose

Test Case 1 – WEIBULL_1.(d)

Expected Results:

```
          Correlation Matrix of Parameter Estimates

     Parameter        BG_DOSE        Slope        Power
     BG_DOSE           1.0000      -0.9947       0.9713
     Slope            -0.9947       1.0000      -0.9874
     Power             0.9713      -0.9874       1.0000
```

Actual Results:

```
Asymptotic Correlation Matrix of Parameter Estimates

                Background_Dose      Slope        Power
Background_Dose              1      -0.99         0.97
          Slope          -0.99          1        -0.99
          Power           0.97      -0.99            1
```

A.7 Gamma Model with Background Dose

Test Case 1 – GAMMA_1.(d)

Expected Results:

```
       Correlation Matrix of Parameter Estimates

       Parameter       BG_DOSE       Slope        Power
       BG_DOSE         1.0000        0.9911       0.9832
       Slope           0.9911        1.0000       0.9520
       Power           0.9832        0.9520       1.0000
```

Actual Results:

```
       Asymptotic Correlation Matrix of Parameter Estimates

              Background_Dose       Slope        Power
Background_Dose            1         0.97         0.99
       Slope            0.97            1         0.99
       Power            0.99         0.99            1
```

APPENDIX B

FORMATS OF BMDS *.(d) FILES FOR EACH MODEL TYPE

In BMDS, the *.(d) files provide all of the user specifications, settings, and data to be input to the computer programs (initially, a main program like *multistage.c*) that perform the computations and that report the results to the *.OUT text file. The formats of the *.(d) files are as follows according to the type of model.

B.1   Multistage and Cancer Models

```
                                              Notes
Multistage                                    1
BMDS MODEL RUN                                2
EXAMPLE.SET                                   3
EXAMPLE                                       4
4 2                                           5  6
250 2.22045e-16 1.49012e-8 1 0 0 1 1   7a 7b 7c 8 9 10 11 12
0.10 0 0.95                                   13  14  15
-9999 -9999 -9999 ...               16  17  18+
0                                             19
-9999 -9999 -9999 ...               20  21  22+
Dose Resp NEGATIVE_RESPONSE         23  24  25
0 3 47
50 6 44                            etc.
100 10 40
150 19 31
```

Explanations to Notes:

1. Model name, in this case, the string Multistage (Could be Cancer)
2. User notes
3. Input file name
4. Output file name
5. Number of Observations
6. Degree of Polynomial
7a. Maximum # of iterations
         = Default of 250 if user does not input a value
         = User input value otherwise
7b. Rel Function Convergence
         = Default of 2.22045e-16 if user does not input a value
         = User input value otherwise
7c. Parameter Convergence
         = Default of 1.49012e-8 if user does not input a value
         = User input value otherwise
8. BMDL Curve Calculation

= 1 if BMDL Curve Calculation box is checked
         = 0 otherwise
9. Restrict Betas >= 0
         = 1 if Restrict Betas >= 0 box is checked(Must be set for type 2)
         = 0 otherwise

10. BMD Calculation
      = 1 if BMD calculation box is checked
      = 0 otherwise
11. Append or Overwrite Output File
      = 1 if Append is selected
      = 0 if Overwrite is selected
      !! NOTE: This parameter is automatically set to 0 by the user interface and can
      only be changed by manually editing the .(d) file, e.g., for batch processing!!

12. Smooth Option
      = 0 if Unique
      = 1 if C-Spline
13. BMR Factor (BMR level)
      = User input value (or default of .100)
14. Risk Type
      = 0 if Extra
      = 1 if Added
15. Confidence Level
      = User input value (or default of .950)
16. Background Parameter
      = User input value if Specified Option is selected
      = -9999 if Specified is not selected or when the user selects the Specified option,
       but does not enter a value
17. Beta1 Parameter
      = User input value if Specified Option is selected

= -9999 if Specified is not selected or when the user selects the Specified option,
      but does not enter a value
18. Beta2 Parameter
      = User input value if Specified Option is selected
      = -9999 if Specified is not selected or when the user selects the Specified option,
      but does not enter a value
18+. Etc. for Beta3, Beta4...
      = User input value if Specified Option is selected
      = -9999 if Specified is not selected or when the user selects the Specified option,

  but does not enter a value
19. Initialize Parameters
      = 1 if one or more parameters are set to initialized
      = 0 otherwise
20. Background Parameter
      = User specified initial value if "initialized" is selected for this parameter
      = -9999 if "initialized" is not selected
21. Beta1 Parameter
      = User specified initial value if "initialized" is selected for this parameter
      = -9999 if "initialized" is not selected
22. Beta2 Parameter
      = User specified initial value if "initialized" is selected for this parameter
      = -9999 if "initialized" is not selected
22+. Etc. for Beta3, Beta4,...as necessary
23. Dose Name
24. Response Name
25. Constant String: NEGATIVE_RESPONSE
etc. Data: Dose in first column
               Response in Second
               Total minus Response in third column

B.2   Logistic and Log-Logistic Models

	Notes
LogisticWithBackgroundResponse	1
BMDS MODEL RUN	2
EXAMPLE.SET	3
EXAMPLE	4
4	5
250 2.22045e-16 1.49012e-8 1 0 0 1 1 0	6a 6b 6c 7 8 9 10 11 12
0.10 0 0.95	13 14 15
-9999 -9999 -9999	16 17 18
0	19
-9999 -9999 -9999	20 21 22
Dose Resp NEGATIVE_RESPONSE	23 24 25
0 3 47	
50 6 44	etc.
100 10 40	
150 19 31	

Explanations to Notes:

1. Model name, in this case, the string LogisticWithBackgroundResponse
    For log-logistic the string will be LogLogisticWithBackgroundDose
2. User notes
3. Input file name
4. Output file name
5. Number of Observations
6a. Maximum # of iterations
        = Default of 250 if user does not input a value
        = User input value otherwise
6b. Rel Function Convergence
        = Default of 2.22045e-16 if user does not input a value
        = User input value otherwise
6c. Parameter Convergence
        = Default of 1.49012e-8 if user does not input a value
        = User input value otherwise
7. BMDL Curve Calculation
        = 1 if BMDL Curve Calculation box is checked
        = 0 otherwise
8. Currently this parameter is ignored
9. Restrict Slope >= 0
        = 1 if Restrict Slope >= 0 box is checked
        = 0 otherwise
10. BMD Calculation
        = 1 if BMD calculation box is checked
        = 0 otherwise
11. Append or Overwrite Output File
        = 1 if Append is selected
        = 0 if Overwrite is selected
        !! NOTE: This parameter is automatically set to 0 by the user interface and can
        only be changed by manually editing the .(d) file, e.g., for batch processing.

12. Smooth Option
        = 0 if Unique
        = 1 if C-Spline

13. BMR Factor (BMR level)
   = User input value (or default of .100)
14. Risk Type
   = 0 if Extra
   = 1 if Added
15. Confidence Level
   = User input value (or default of .950)
16. Background Parameter
   = User input value if Specified Option is selected
   = -9999 if Specified is not selected or when the user selects the Specified option,
      but does not enter a value or if Log transformation not selected
17. Slope Parameter
   = User input value if Specified Option is selected
   = -9999 if Specified is not selected or when the user selects the Specified option,
      but does not enter a value
18. Intercept Parameter
   = User input value if Specified Option is selected
   = -9999 if Specified is not selected or when the user selects the Specified option,
      but does not enter a value
19. Initialize Parameters
   = 1 if one or more parameters are set to initialized
   = 0 otherwise
20. Background Parameter
   = User specified initial value if "initialized" is selected for this parameter
   = -9999 if "initialized" is not selected or if Log transformation not selected
   *For logistic model with background response, this parameter is background response; For log-logistic model with background dose, this parameter is background dose
21. Slope Parameter
   = User specified initial value if "initialized" is selected for this parameter
   = -9999 if "initialized" is not selected
22. Intercept Parameter
   = User specified initial value if "initialized" is selected for this parameter
   = -9999 if "initialized" is not selected

23. Dose Name
24. Response Name
25. Constant String: NEGATIVE_RESPONSE

etc. Data: Dose in first column
               Response in Second
               Total minus Response in third column

B.3   Probit and Log-Probit Models

ProbitWithBackgroundResponse   1
BMDS MODEL RUN   2
EXAMPLE.SET   3
EXAMPLE   4
4   5
250 2.22045e-16 1.49012e-8 1 0 0 1 1 0 6a 6b 6c 7 8 9 10 11 12
0.10 0 0.95   13 14 15
-9999 -9999 -9999   16 17 18
0   19
-9999 -9999 -9999   20 21 22
Dose Resp NEGATIVE_RESPONSE   23 24 25
0 3 47
50 6 44   etc.
100 10 40
150 19 31

Explanations to Notes:

1. Model name, in this case, the string ProbitWithBackgroundResponse
   For log-probit the string will be LogProbitWithBackgroundDose
2. User notes
3. Input file name
4. Output file name
5. Number of Observations
6a. Maximum # of iterations
        = Default of 250 if user does not input a value
        = User input value otherwise
6b. Rel Function Convergence
        = Default of 2.22045e-16 if user does not input a value
        = User input value otherwise
6c. Parameter Convergence
        = Default of 1.49012e-8 if user does not input a value
        = User input value otherwise
7. BMDL Curve Calculation
        = 1 if BMDL Curve Calculation box is checked
        = 0 otherwise
8. Currently this parameter is ignored
9. Restrict Slope
        = 1 if Restrict Slope >= 1 box is checked
        = 0 otherwise
10. BMD Calculation
        = 1 if BMD calculation box is checked
        = 0 otherwise
11. Append or Overwrite Output File
        = 1 if Append is selected
        = 0 if Overwrite is selected
        !! NOTE: This parameter is automatically set to 0 by the user interface and can
        only be changed by manually editing the .(d) file, e.g., for batch processing!!

12. Smooth Option
        = 0 if Unique
        = 1 if C-Spline

13. BMR Factor (BMR level)
      = User input value (or default of .100)
14. Risk Type
      = 0 if Extra
      = 1 if Added
15. Confidence Level
      = User input value (or default of .950)
16. Background Parameter
      = User input value if Specified Option is selected
      = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value or if Log transformation not selected

   *For Probit model with background response, this parameter is background response; For log-probit model with background dose, this parameter is background dose

17. Slope Parameter
      = User input value if Specified Option is selected
      = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value
18. Intercept Parameter
      = User input value if Specified Option is selected
      = -9999 if Specified is not selected or when the user selects the Specified option, does not enter a value
19. Initialize Parameters
      = 1 if one or more parameters are set to initialized
      = 0 otherwise
20. Background Parameter
      = User specified initial value if "initialized" is selected for this parameter
      = -9999 if "initialized" is not selected or if Log transformation not selected

   *For Probit model with background response, this parameter is background response; For log-probit model with background dose, this parameter is background dose

21. Slope Parameter
      = User specified initial value if "initialized" is selected for this parameter
      = -9999 if "initialized" is not selected
22. Intercept Parameter
      = User specified initial value if "initialized" is selected for this parameter
      = -9999 if "initialized" is not selected

23. Dose Name
24. Response Name
25. Constant String: NEGATIVE_RESPONSE

etc. Data: Dose in first column
              Response in Second
              Total minus Response in third column

B.4 Weibull Model

WeibullWithBackgroundResponse	1
BMDS MODEL RUN	2
EXAMPLE.SET	3
EXAMPLE	4
4	5
250 2.22045e-16 1.49012e-8 1 1 1 1 0	6a 6b 6c 7 8 9 10 11
0.10 0 0.95	12 13 14
-9999 -9999 -9999	15 16 17
0	18
-9999 -9999 -9999	19 20 21
Dose Resp NEGATIVE_RESPONSE	22 23 24
0 3 47	
50 6 44	etc.
100 10 40	
150 19 31	

Explanations to Notes:

1. Model name, in this case, the string WeibullWithBackgroundResponse
2. User notes
3. Input file name
4. Output file name
5. Number of Observations
6a. Maximum # of iterations
     = Default of 250 if user does not input a value
     = User input value otherwise
6b. Rel Function Convergence
     = Default of 2.22045e-16 if user does not input a value
     = User input value otherwise
6c. Parameter Convergence.
     = Default of 1.49012e-8 if user does not input a value
     = User input value otherwise
7. BMDL Curve Calculation
     = 1 if BMDL Curve Calculation box is checked
     = 0 otherwise
8. Restrict power >= 1
     = 1 if Restrict Power >= 1 box is checked
     = 0 otherwise
9. BMD Calculation
     = 1 if BMD calculation box is checked
     = 0 otherwise
10. Append or Overwrite Output File
     = 1 if Append is selected
     = 0 if Overwrite is selected
     !! NOTE: This parameter is automatically set to 0 by the user interface and can only be changed
by manually editing the .(d) file, e.g., for batch processing!!
11. Smooth Option
     = 0 if Unique
     = 1 if C-Spline
12. BMR Factor (BMR level)
     = User input value (or default of .100)

13. Risk Type

     = 0 if Extra

     = 1 if Added

14. Confidence Level

     = User input value (or default of .950)

15. Background Dose Parameter

     = User input value if Specified Option is selected

     = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value

16. Slope Parameter

     = User input value if Specified Option is selected

     = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value

17. Power parameter

     = User input value if Specified Option is selected

     = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value

18. Initialize Parameters

     = 1 if one or more parameters are set to initialized

     = 0 otherwise

19. Background Dose Parameter.

     = User specified initial value if "initialized" is selected for this parameter

     = -9999 if "initialized" is not selected

20. Slope Parameter

     = User specified initial value if "initialized" is selected for this parameter

     = -9999 if "initialized" is not selected

21. Power Parameter

     = user specified initial value if "initialized" is selected for this parameter

     = -9999 if "initialized" is not selected

22. Dose Name

23. Response Name

24. Constant String: NEGATIVE_RESPONSE

etc. Data: Dose in first column

               Response in Second

               Total minus Response in third column

B.5   Gamma Model

	Notes
GammaWithBackgroundDose	1
BMDS MODEL RUN	2
EXAMPLE.SET	3
EXAMPLE	4
4	5
250 2.22045e-16 1.49012e-8 1 1 1 1 0	6a 6b 6c 7 8 9 10 11
0.10 0 0.95	12 13 14
-9999 -9999 -9999	15 16 17
0	18
-9999 -9999 -9999	19 20 21
Dose Resp NEGATIVE_RESPONSE	22 23 24
0 3 47	
50 6 44	etc.
100 10 40	
150 19 31	

Explanations to Notes:

1. Model name, in this case, the string GammaWithBackgroundDose
2. User notes
3. Input file name
4. Output file name

5. Number of Observations
6a. Maximum # of iterations
    = Default of 250 if user does not input a value
    = User input value otherwise
6b. Rel Function Convergence
    = Default of 2.22045e-16 if user does not input a value
    = User input value otherwise
6c. Parameter Convergence.
    = Default of 1.49012e-8 if user does not input a value
    = User input value otherwise
7. BMDL Curve Calculation
    = 1 if BMDL Curve Calculation box is checked
    = 0 otherwise
8. Restrict Power >= 1
    = 1 if Restrict Power >= 1 box is checked
    = 0 otherwise
9. BMD Calculation
    = 1 if BMD calculation box is checked
    = 0 otherwise
10. Append or Overwrite Output File
    = 1 if Append is selected
    = 0 if Overwrite is selected
    !! NOTE: This parameter is automatically set to 0 by the user interface and can only be changed
    by manually editing the .(d) file, e.g., for batch processing!!
11. Smooth Option
    = 0 if Unique
    = 1 if C-Spline
12. BMR Factor (BMR level)
    = User input value (or default of .100)

13. Risk Type
    = 0 if Extra
    = 1 if Added
14. Confidence Level
    = User input value (or default of .950)
15. Background Dose Parameter
    = User input value if Specified Option is selected
    = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value
16. Slope Parameter
    = User input value if Specified Option is selected
    = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value
17. Power Parameter
    = User input value if Specified Option is selected
    = -9999 if Specified is not selected or when the user selects the Specified option, but does not enter a value
18. Initialize Parameters
    = 1 if one or more parameters are set to initialized
    = 0 otherwise
19. Background Dose Parameter.
    = User specified initial value if "initialized" is selected for this parameter
    = -9999 if "initialized" is not selected
20. Slope Parameter
    = User specified initial value if "initialized" is selected for this parameter
    = -9999 if "initialized" is not selected
21. Power Parameter
    = User specified initial value if "initialized" is selected for this parameter
    = -9999 if "initialized" is not selected
22. Dose Name
23. Response Name
24. Constant String: NEGATIVE_RESPONSE

etc. Data: Dose in first column
                        Response in Second
                        Total minus Response in third column

APPENDIX C

CHANGES TO BMDS FILE SET

Appendix C documents changes made to source code (.c and .h files) and some helper .bat files. Changes to object files and executables are not included.

C.1   Multistage and Cancer Models

The folding files have been added or edited to the BMDS file set.

Multistage_bg_dose folder
> This folder is new but based on the Multistage folder in BMDS-1-4-20051110.

> Modified in this release:
>> multistage_fun.h   header file for multistage function

>> multistage_fun.c   Objective function and non linear constraint functions based on BMR and Log Likelihood used in calculating BMDL were added.

>> multistage_bgdose.c   Background dose parameter (alpha) was added to the model for parameter fitting and BMDL calculations. All optimizations calls for Maximum Likelihood and BMDL calculations were changed to use donlp3 (C Version), instead of donlp2.

>> test_multistage.bat batch file to run a suite of test cases. These are validation runs to compare with the output from SAS.

>> ***.(d) A set of test files is included

>> The files Proflik.c and PROBLEM.INC are unchanged from the Multistage folder. Files from the multistage, not required for Multistage with background does are not included in this folder.

Cancer_bg_dose Folder
> This folder is new, but based on the Cancer folder in BMDS-1-4-20051110:
>> cancer_fun.h   header file for multistage function

>> cancer_fun.c   Objective function and non linear constraint functions based on BMR and Log Likelihood used in calculating BMDL were added.

cancer_bgdose.c   Background dose parameter (alpha) was added to the
        model for parameter fitting and BMDL calculations. All optimizations
        calls for Maximum Likelihood and BMDL calculations were changed
        to use donlp3 (C Version), instead of donlp2. This file replace
        cancer.c

test_cancer.bat   batch file to run a suite of test cases. These are
validation runs to compare with the output from SAS

.

***.(d) A set of test files is included

The files Proflik.c and PROBLEM.INC are unchanged from the Cancer
moder

Files from the Cancer folder, not required for Cancer with Background
Dose Parameter are not included in this folder.

Assist Folder
    Files:
        Modified:
            confint.h   header file for BMD and BMDL calculations
            confint.c   source code for BMD and BMDL calculations

    Directories:
        Modified:
        /donlp3   This is the donlp2 rewritten for C. Some very minor changes
            were made to return needed data in the necessary form. No changes
            were made to the computational algorithms. The revised
            multistage_bgdose.c only uses donlp3 and makes no reference to the
            FORTRAN version of donlp2.

        Modified:
            donlp3.fun.c Implementation of interface to donlp3 used for
                maximum likelihood and BMDL calculations

            donlp3fun.h Header file for interface to donlp3 used for maximum
                likelihood and BMDL calculations

            donlp3return.h Header file for defining structure that returns from
                donlp3 call

C.2   Logistic and Log-Logistic Models

The following folders and member files have been added to the BMDS file set.

Logistic_bg_response folder
  With files:
        logistic_fun.h   header file for logistic function

        logistic_fun.c   Objective function and non linear constraint functions
              based on BMR and Log Likelihood used in calculating BMDL were
              added.

        logistic_bg_response.c   Background response parameter (alpha) was
              added to the model for parameter fitting and BMDL calculations.

        test.bat   batch file to run a suite of test cases. These are validation
        runs to compare with the output from SAS

        ***.(d) A set of test files is included

Log_logistic_bg_dose folder
  With files:
        Log_logistic_fun.h   header file for log-logistic function

        Log_logistic_fun.c   Objective function and non linear constraint
              functions based on BMR and Log Likelihood used in calculating
              BMDL were added.

        Log_logistic_bg_dose.c   Background response parameter (alpha) was
              changed to background dose parameter to the model for parameter
              fitting and BMDL calculations.

        test.bat   batch file to run a suite of test cases. These are validation
        runs to compare with the output from SAS

        ***.(d) A set of test files is included

Changes to the Assist folder that are listed in Section D.1 were also made for the
logistic and log-logistic modules.

C.3   Probit and Log-Probit Models

The following folders and member files have been added to the BMDS file set.

Probit_bg_response folder
  With files:
        probit_fun.h -- Header file for probit function

probit_fun.c -- Objective function and non linear constraint functions based on BMR and Log Likelihood used in calculating BMDL were added.

probit_bg_response.c -- Background response parameter (alpha) was added to the model for parameter fitting and BMDL calculations.

test.bat -- Batch file to run a suite of test cases. These are validation runs to compare with the output from SAS

***.(d) -- A set of test files is included

Log_probit_bg_dose folder
  With files:
        Log_probit_fun.h -- Header file for log-probit function

        Log_probit_fun.c -- Objective function and non linear constraint functions based on BMR and Log Likelihood used in calculating BMDL were added.

        Log_probit_bg_dose.c -- Background response parameter (alpha) was changed to background dose parameter to the model for parameter fitting and BMDL calculations.

        test.bat -- Batch file to run a suite of test cases. These are validation runs to compare with the output from SAS

        ***.(d) -- A set of test files is included

In the Assist folder, the following changes were made:

    Files:
        Modified:
            confint.h -- Header file for BMD and BMDL calculations
                Two new global data structures g_s_lower_limit and g_s_upper_limit were added to hold upper and lower bounds on parameters. The bounds are now set in individual models (Probit, Log-probit etc.)
            confint.c -- Source code for BMD and BMDL calculations

    Directories:
        Modified:
            /donlp3 -- This is the donlp2 rewritten for C. The revised Probit and Log-probit only uses donlp3 and makes no reference to run_dmngb files for optimization purposes.

Modified:

donlp3.fun.c -- Implementation of interface to donlp3 used for maximum likelihood and BMDL calculations. The limits on parameters low[i] and up[i] are now set according to values passed in the global data structures g_s_lower_limit and g_s_upper_limit.

donlp3fun.h -- Header file for interface to donlp3 used for maximum likelihood and BMDL calculations

donlp3return.h -- Header file for defining structure that returns from donlp3 call

C.4   Weibull Model

The following folders and member files have been added to the BMDS file set.

Weibull_bg_dose folder
    With files:

weibull_fun.h   -- Header file for Weibull function

weibull_fun.c   -- Objective function and non linear constraint functions based on BMR and Log Likelihood used in calculating BMDL were added.

weibull_bgdose.c -- Background response parameter (alpha) was added to the model for parameter fitting and BMDL calculations.

test.bat -- Batch file to run a suite of test cases. These are validation runs to compare with the output from SAS

***.(d) -- A set of test files is included

Changes to the Assist folder that are listed in Section D.3 were also made for the Weibull module.

C.5   Gamma Model

The following folders and member files have been added to the BMDS file set.

Gamma_bg_dose folder
    With files:

gamma_fun.h   -- Header file for Gamma function

gamma_fun.c   -- Objective function and non linear constraint functions based on BMR and Log Likelihood used in calculating BMDL were added.

gamma_bgdose.c -- Background response parameter (alpha) was added to the model for parameter fitting and BMDL calculations.

test.bat -- Batch file to run a suite of test cases. These are validation runs to compare with the output from SAS

***.(d) -- A set of test files is included

In the Assist folder, two files were modified to accommodate the analytical differentiation of Incomplete Gamma Integral in covariance calculations.

Directories:
Modified:
/dcdflib/cdflib.h – Header file for functions to evaluate cumulative distribution functions for all the models.
Function header trigam(double *d) was added to evaluate Trigamma

/dcdflib/dcdflib.c – Source code for evaluating CDF
Code for implementing Trigamma evaluation was added.

Changes to the Assist folder that are listed in Section D.3 were also made for the gamma module.

APPENDIX D

OUTPUT FILES GENERATED BY EXECUTING THE NEW MODULES

D.1   Multistage Model

```
====================================================================
           Multistage Model with Background Dose. (Version: 2.8;  Date: 12/01/2006)
           Input Data File: MULTISTAGE01_14.(d)
           Gnuplot Plotting File:  MULTISTAGE01_14.plt
                                              Wed Apr 18 10:10:27 2007
====================================================================
```

BMDS MODEL RUN
~~~~~~

   The form of the probability function is:

   P[response] = [1-EXP(
          -beta1*(dose + alpha)^1-beta2*(dose+alpha)^2-beta3*(dose+alpha)^3)]

   The parameter betas are restricted to be positive

   Dependent variable = EFFECT1
   Independent variable = DOSE

Total number of observations = 5
Total number of records with missing values = 0
Total number of parameters in model = 4
Total number of specified parameters = 0
Degree of polynomial = 3

Maximum number of iterations = 250
Relative Function Convergence has been set to: 1e-008
Parameter Convergence has been set to: 1e-008

            Default Initial Parameter Values
          Background_Dose =   0.5
               Beta(1) =   0.0393758
               Beta(2) =   0
               Beta(3) =   2.1377

      Asymptotic Correlation Matrix of Parameter Estimates

    (*** The model parameter(s) -Beta(2)
         have been estimated at a boundary point, or have been specified by the user,
         and do not appear in the correlation matrix)

        Background_Dose   Beta(1)   Beta(3)

 Background_Dose   1   -0.95   -0.83

       Beta(1)   -0.95   1   0.65

       Beta(3)   -0.83   0.65   1

             Parameter Estimates

```
                            95.0% Wald Confidence Interval
      Variable      Estimate     Std. Err.    Lower Conf. Limit    Upper Conf. Limit
  Background_Dose      0.440805    0.639246       -0.812094            1.6937
       Beta(1)        0.320712    0.760817       -1.17046             1.81189
       Beta(2)           0        NA
       Beta(3)        0.468105    0.454821       -0.423329            1.35954
```

NA - Indicates that this parameter has hit a bound
   implied by some inequality constraint and thus
   has no standard error.

                    Analysis of Deviance Table

     Model   Log(likelihood) # Param's Deviance Test d.f. P-value
   Full model   -27.2023   5
   Fitted model   -27.5963   3   0.788118   2   0.6743
   Reduced model -34.6574   1   14.9102   4   0.004891

      AIC: 61.1926

                        Goodness of Fit
                                     Scaled
      Dose   Est._Prob.   Expected Observed   Size   Residual
   ----
    0.0000   0.1660   1.660   2   10   0.289
    0.2000   0.2801   2.801   3   10   0.140
    0.5000   0.4992   4.992   4   10   -0.627
    0.8000   0.7253   7.253   7   10   -0.179
    1.0000   0.8447   8.447   9   10   0.483

   Chi^2 = 0.76   d.f. = 2   P-value = 0.6830

   Benchmark Dose Computation

Specified effect =   0.1

Risk Type   =   Extra risk

Confidence level =   0.95

       BMD =   0.151004

       BMDL =   0.0612734

       BMDU =   0.20703
```

## D.2    Cancer Model

```
==================================================================
           Cancer Model with Background Dose. (Version: 1.5;  Date: 12/01/2006)
           Input Data File: CANCER01_2.(d)
           Gnuplot Plotting File:  CANCER01_2.plt
                                              Wed Apr 18 08:48:06 2007
==================================================================
```

BMDS MODEL RUN
~~~~~~

 The form of the probability function is:

 P[response] = [1-EXP(
          -beta1*(dose + alpha)^1-beta2*(dose+alpha)^2)]

 The parameter betas are restricted to be positive

 Dependent variable = EFFECT1
 Independent variable = DOSE

Total number of observations = 5
Total number of records with missing values = 0
Total number of parameters in model = 3
Total number of specified parameters = 0
Degree of polynomial = 2

Maximum number of iterations = 250
Relative Function Convergence has been set to: 1e-008
Parameter Convergence has been set to: 1e-008

             Default Initial Parameter Values
            Background_Dose =   0
                 Beta(1) =   0.29471
                 Beta(2) =   1.85687

        Asymptotic Correlation Matrix of Parameter Estimates

         Background_Dose   Beta(1)   Beta(2)

 Background_Dose   1   -0.97   -0.22

     Beta(1)   -0.97   1   1.8e-007

     Beta(2)   -0.22   1.8e-007   1

                    Parameter Estimates

                         95.0% Wald Confidence Interval
   Variable   Estimate   Std. Err.   Lower Conf. Limit   Upper Conf. Limit
 Background_Dose   0.5443   1.26725   -1.93946   3.02806
     Beta(1)   7.84436e-007   1.7787   -3.48618   3.48619
     Beta(2)   0.721278   0.427539   -0.116682   1.55924

              Analysis of Deviance Table

   Model   Log(likelihood) # Param's   Deviance   Test d.f.   P-value
   Full model   -27.8237   5
```

```
Fitted model      -28.6199     3     1.59241    2        0.451
Reduced model      -34.6173     1     13.5872    4        0.008736
```

AIC: 63.2399

                    Goodness of Fit
                                    Scaled
    Dose   Est._Prob.   Expected Observed   Size   Residual

   0.0000   0.1924   1.924   2   10   0.061
   0.2000   0.3294   3.294   4   10   0.475
   0.5000   0.5446   5.446   4   10   -0.918
   0.8000   0.7284   7.284   7   10   -0.202
   1.0000   0.8210   8.210   9   10   0.652

 Chi^2 = 1.54   d.f. = 2   P-value = 0.4634

   Benchmark Dose Computation

Specified effect =   0.1

Risk Type   =   Extra risk

Confidence level =   0.95

       BMD =   0.120784

      BMDL =   0.0542246

      BMDU =   0.183636

Taken together, (0.0542246, 0.183636) is a 90   % two-sided confidence
interval for the BMD

Cancer Slope Factor =   1.84418

D.3   Logistic Model

```
===================================================================
            Logistic Model With Background Response. (Version: 2.9; Date: 12/01/2006)
            Input Data File: LOGISTIC01_19.(d)
            Gnuplot Plotting File:  LOGISTIC01_19.plt
                                                Wed Apr 18 10:06:02 2007
===================================================================
```

 BMDS MODEL RUN
~~~~~~

  The form of the probability function is:

  P[response] = background+(1-background)/[1+EXP(-intercept-slope*dose)]

  Dependent variable = EFFECT1
  Independent variable = DOSE
  Slope parameter is not restricted

  Total number of observations = 5
  Total number of records with missing values = 0
  Maximum number of iterations = 250
  Relative Function Convergence has been set to: 1e-008
  Parameter Convergence has been set to: 1e-008

              Default Initial Parameter Values
        background_response =   0
              intercept =   -1.56202
                  slope =   0.846976

        Asymptotic Correlation Matrix of Parameter Estimates

            background_response   intercept   slope

background_response   1   -0.87   0.74

      intercept   -0.87   1   -0.92

          slope   0.74   -0.92   1

              Parameter Estimates

                        95.0% Wald Confidence Interval
     Variable   Estimate   Std. Err.   Lower Conf. Limit Upper Conf. Limit
background_response 0.102125   0.176216   -0.243251   0.447502
      intercept   -2.59769   1.80274   -6.13099   0.935606
          slope   1.23048   0.612038   0.0309098   2.43006

              Analysis of Deviance Table

    Model   Log(likelihood) # Param's Deviance Test d.f. P-value
    Full model   -24.993   5
    Fitted model   -25.3817   3   0.777418   2   0.6779
    Reduced model   -34.0146   1   18.0432   4   0.00121

       AIC: 56.7635

                  Goodness of Fit

```
                        Scaled
  Dose    Est._Prob.  Expected  Observed  Size    Residual
-------------------------------------------------------------------------
  0.0000    0.1643     1.643      2        10      0.304
  0.5000    0.2108     2.108      2        10     -0.084
  1.0000    0.2845     2.845      2        10     -0.592
  2.0000    0.5204     5.204      6        10      0.504
  4.0000    0.9200     9.200      9        10     -0.233
```

Chi^2 = 0.76   d.f. = 2   P-value = 0.6845

  Benchmark Dose Computation

Specified effect =   0.1

Risk Type   =   Extra risk

Confidence level =   0.95

        BMD =   0.77767

       BMDL =   0.409144

       BMDU =   1.9538

D.4   Log-Logistic Model

```
=====================================================================
          LogLogistic_Background_Dose Model. (Version: 1.2; Date: 12/03/2006)
          Input Data File: LOGLOGISTIC01_20.(d)
          Gnuplot Plotting File:  LOGLOGISTIC01_20.plt
                                              Wed Apr 18 10:05:24 2007
=====================================================================
```

BMDS MODEL RUN
~~~~~~

  The form of the probability function is:

  P[response] = 1/[1+EXP(-intercept-slope*Log(dose + background))]

  Dependent variable = EFFECT1
  Independent variable = DOSE
  Slope parameter is not restricted

  Total number of observations = 5
  Total number of records with missing values = 0
  Maximum number of iterations = 250
  Relative Function Convergence has been set to: 1e-008
  Parameter Convergence has been set to: 1e-008

  User has chosen the log transformed model

              Default Initial Parameter Values
            background_dose =   0
                 intercept =   -0.375342
                     slope =   1.66797

        Asymptotic Correlation Matrix of Parameter Estimates

            background_dose   intercept   slope

background_dose   1   -0.57   0.43

    intercept   -0.57   1   -0.55

       slope   0.43   -0.55   1

                    Parameter Estimates

                              95.0% Wald Confidence Interval
    Variable   Estimate   Std. Err.   Lower Conf. Limit   Upper Conf. Limit
background_dose   0.00313606   0.135313   -0.262073   0.268345
    intercept   -0.421533   0.493474   -1.38872   0.545658
       slope   1.59713   0.596039   0.428913   2.76534

                    Analysis of Deviance Table

    Model   Log(likelihood) # Param's Deviance Test d.f.   P-value
    Full model   -21.7151   5
    Fitted model   -21.9048   3   0.379458   2   0.8272
    Reduced model   -34.0146   1   24.599   4   <.0001

       AIC: 49.8096

```
              Goodness  of  Fit
                                 Scaled
    Dose    Est._Prob.   Expected   Observed   Size    Residual
-------------------------------------------------------------------------
   0.0000    0.0001      0.001       0        10     -0.026
   0.5000    0.1797      1.797       2        10      0.167
   1.0000    0.3973      3.973       4        10      0.017
   2.0000    0.6655      6.655       6        10     -0.439
   4.0000    0.8574      8.574       9        10      0.385

 Chi^2 = 0.37     d.f. = 2      P-value = 0.8310


   Benchmark Dose Computation

Specified effect =        0.1

Risk Type      =     Extra risk

Confidence level =        0.95

        BMD =      0.325965

       BMDL =      0.0662691

       BMDU =       0.630337
```

D.5   Probit Model

```
=====================================================================
            Log Probit Model with Background Dose. (Version: 1.1;  Date: 12/03/2006)
            Input Data File: LOGPROBIT_1.(d)
            Gnuplot Plotting File:  LOGPROBIT_1.plt
                                                    Wed Apr 18 09:45:00 2007
=====================================================================
```

 BMDS MODEL RUN
~~~~~~

 The form of the probability function is:

 P[response] = CumNorm(Intercept+Slope*Log(Dose + Background)),

 where CumNorm(.) is the cumulative normal distribution function

 Dependent variable = EFFECT1
 Independent variable = DOSE
 Slope parameter is not restricted

 Total number of observations = 5
 Total number of records with missing values = 0
 Maximum number of iterations = 250
 Relative Function Convergence has been set to: 1e-008
 Parameter Convergence has been set to: 1e-008

 User has chosen the log transformed model

            Default Initial (and Specified) Parameter Values
            background_dose =   0
                intercept =   -0.41031
                    slope =   1.0769

        Asymptotic Correlation Matrix of Parameter Estimates

            background_dose   intercept   slope

 background_dose   1   -0.99   0.97

     intercept   -0.99   1   -0.99

        slope   0.97   -0.99   1

                    Parameter Estimates

                                95.0% Wald Confidence Interval
        Variable   Estimate   Std. Err.   Lower Conf. Limit   Upper Conf. Limit
 background_dose   0.529767   1.67131   -2.74594   3.80547
        intercept   -1.23675   2.71804   -6.564   4.0905
            slope   1.61284   1.67091   -1.66209   4.88777

                    Analysis of Deviance Table

    Model   Log(likelihood) # Param's Deviance Test d.f. P-value
    Full model   -19.989   5
    Fitted model   -20.6168   3   1.25558   2   0.5338
    Reduced model   -33.2032   1   26.4284   4   <.0001

AIC: 47.2336

        Goodness of Fit
                            Scaled
   Dose   Est._Prob.   Expected   Observed   Size   Residual

   0.0000   0.0119   0.119   0   10   -0.347
   0.5000   0.1171   1.171   2   10   0.815
   1.0000   0.2908   2.908   2   10   -0.632
   2.0000   0.6026   6.026   6   10   -0.017
   4.0000   0.8849   8.849   9   10   0.150

 Chi^2 = 1.21   d.f. = 2   P-value = 0.5470

   Benchmark Dose Computation

Specified effect =   0.1

Risk Type   =   Added risk

Confidence level =   0.95

        BMD =   0.482766

       BMDL =   0.169529

       BMDU =   0.89335

D.7 Weibull Model

```
=====================================================================
        Weibull Model with Background Dose (Version: 1.1;  Date: 12/06/2006)
        Input Data File: WEIBULL01.(d)
        Gnuplot Plotting File:  WEIBULL01.plt
                                                Wed Apr 18 10:08:03 2007
=====================================================================
```

BMDS MODEL RUN
~~~~~~

 The form of the probability function is:

 P[response] = [1-EXP(-slope*(dose+background)^power)]

 Dependent variable = EFFECT1
 Independent variable = DOSE
 Power parameter is not restricted

 Total number of observations = 5
 Total number of records with missing values = 0
 Maximum number of iterations = 250
 Relative Function Convergence has been set to: 1e-008
 Parameter Convergence has been set to: 1e-008

           Default Initial (and Specified) Parameter Values
            Background_Dose =   0.0192228
                    Slope =   0.05787
                    Power =   1.2038

        Asymptotic Correlation Matrix of Parameter Estimates

           Background_Dose   Slope   Power

 Background_Dose   1   -0.99   0.97

        Slope   -0.99   1   -0.99

        Power   0.97   -0.99   1

                      Parameter Estimates

                                95.0% Wald Confidence Interval
     Variable   Estimate   Std. Err.   Lower Conf. Limit Upper Conf. Limit
 Background_Dose   0.939874   1.59442   -2.18514   4.06488
        Slope   0.0128504   0.0352123   -0.0561645   0.0818652
        Power   2.07695   1.34617   -0.561504   4.7154

                   Analysis of Deviance Table

    Model   Log(likelihood) # Param's Deviance Test d.f. P-value
   Full model   -134.664   5
  Fitted model   -134.695   3   0.0615643   2   0.9697
  Reduced model   -162.541   1   55.7553   4   <.0001

      AIC: 275.389

```
                Goodness  of  Fit
                                    Scaled
    Dose    Est._Prob.   Expected  Observed   Size    Residual
  -------------------------------------------------------------------
   0.0000   0.0112      1.123      1      100      -0.117
   0.5000   0.0270      2.703      3      100       0.183
   1.0000   0.0496      4.961      5      100       0.018
   2.0000   0.1137     11.368     11      100      -0.116
   4.0000   0.2985     29.854     30      100       0.032

  Chi^2 = 0.06     d.f. = 2      P-value = 0.9695


   Benchmark Dose Computation

Specified effect =        0.1

Risk Type      =    Extra risk

Confidence level =        0.95

        BMD =     1.95249

        BMDL =    1.41095

        BMDU =    2.40362
```

D.8   Gamma Model

```
=====================================================================
          Gamma Model with Background Dose (Version: 1.2;  Date: 12/06/2006)
          Input Data File: GAMMA01.(d)
          Gnuplot Plotting File:  GAMMA01.plt
                                            Wed Apr 18 09:40:22 2007
=====================================================================
```

 BMDS MODEL RUN
 ~~~~~~

   The form of the probability function is:

   P[response]= CumGamma[slope*(dose+background),power],
   where CumGamma(.) is the cummulative Gamma distribution function

   Dependent variable = EFFECT1
   Independent variable = DOSE
   Power parameter is not restricted

   Total number of observations = 5
   Total number of records with missing values = 0
   Maximum number of iterations = 250
   Relative Function Convergence has been set to: 1e-008
   Parameter Convergence has been set to: 1e-008

            Default Initial (and Specified) Parameter Values
            Background_Dose =   4.00042
                    Slope =   0.622059
                    Power =   6.15594

        Asymptotic Correlation Matrix of Parameter Estimates

          Background_Dose   Slope   Power

 Background_Dose   1   0.97   0.99

       Slope   0.97   1   0.99

       Power   0.99   0.99   1

                    Parameter Estimates

                              95.0% Wald Confidence Interval
       Variable   Estimate   Std. Err.   Lower Conf. Limit   Upper Conf. Limit
 Background_Dose   1.73929   3.44251   -5.00791   8.48649
          Slope   0.324779   0.462667   -0.582032   1.23159
          Power   2.73979   4.22511   -5.54126   11.0209

                    Analysis of Deviance Table

     Model   Log(likelihood) # Param's   Deviance Test d.f. P-value
    Full model   -175.136   5
   Fitted model   -175.165   3   0.0576847   2   0.9716
  Reduced model   -200.658   1   51.0434   4   <.0001

        AIC: 356.33

                    Goodness of Fit
```

```
                           Scaled
  Dose    Est._Prob.  Expected   Observed   Size    Residual
----------------------------------------------------------------
 0.0000    0.0319      3.185        3       100      -0.106
 0.5000    0.0568      5.677        6       100       0.139
 1.0000    0.0881      8.806        9       100       0.068
 2.0000    0.1653     16.534       16       100      -0.144
 4.0000    0.3480     34.805       35       100       0.041
```

 Chi^2 = 0.06   d.f. = 2   P-value = 0.9716

  Benchmark Dose Computation

Specified effect =   0.1

Risk Type   =   Extra risk

Confidence level =   0.95

        BMD =   1.55158

        BMDL =   1.00754

        BMDU =   2.18718

APPENDIX E

SAS® Validation Code

This appendix contains the original Weibull model SAS® code as published in Wheeler (2005). The author also kindly provided code for the other modules referenced in his paper. This code is not reproduced here; rather the versions of the code used for validation in this document are listed in this appendix.

E.1 Wheeler's Weibull Model SAS® code (taken from Wheeler (2005).

```
data EGdata;
input dose obs n;
cards;
     0 2 10
     0.5 2 10
     1.0 2 10
     2.0 6 10
     4.0 9 10
;

%LET BMR = 0.1;

PROC NLMIXED data= EGdata;
     PARMS _GAMMA = 0.006 _BDOSE = 0.5 _ALPHA = 1;
     BOUNDS _GAMMA >= 0, _GAMMA <= 1, _ALPHA >= 1, _BDOSE >= 0;
     P = _GAMMA;
     IF (DOSE > 0) THEN DO;
            _LINK = _BDOSE * (DOSE**_ALPHA);
            P = _GAMMA + (1-_GAMMA)*(1-EXP(-_LINK));
     END;
     _X = -LOG(1-&BMR);

     CALL SYMPUT("BMD",_BMD);
     MODEL OBS ~ BINOMIAL(N,P);
RUN;

%macro boundBMD(BMDL,MLIKE,CL);
%let BMR = 0.1;
%let MLIKE = %sysevalf(&mlike/2);
data _temp_;
     val = cinv(1-2*(1-&CL),1)*0.5;
     call symput("CRITVAL",val);
run;
%LET CRITLIKE = &MLIKE;
%DO %WHILE (%SYSEVALF(&CRITLIKE - &MLIKE < &CRITVAL));
     *set up the initial parameters for the new likelihood;
     %LET FIRSTTOKEN = 1;
     %LET LASTBMDL = &BMDL;
     %LET BMDL = %SYSEVALF(0.98*&BMDL);
```

```
%LET BOUNDS = _GAMMA >= 0, _GAMMA <= 1, _ALPHA >= 1,_ALPHA <= 18;
%LET SMODEL = ;
data pe; set ParameterEstimates; if(Parameter='_BDOSE')then delete;
run;
data pe;
        set pe;
run;
ods listing close;
ods output fitstatistics = fitstatistics
                ParameterEstimates=ParameterEstimates;
*fit this new "constrained" likelihood;
proc nlmixed data= one;
        parms /data=pe; ;
        bounds &bounds;

        _X = (-LOG(1-&BMR))**(1/_ALPHA);
        *solve for the BETA parameter BDOSE as a function of the BMD;
        _BDOSE = (_X/&BMDL)**_ALPHA;
        P = _GAMMA;
        IF (DOSE > 0) THEN DO;
                _LINK = _BDOSE*DOSE**_ALPHA;
                P = _GAMMA + (1-_GAMMA)*(1-EXP(-_LINK));
        END;
        MODEL OBS ~ BINOMIAL(N,P);
run;
ods listing;
*obtain the Fit statistics to determine if the algorithm has bounded
the BMDL;
data fitstatistics;
        set fitstatistics;
        format value best16.;
        informat value best16.;
        if (Descr = "-2 Log Likelihood");

        keep value NegLogLike;
run;
data _temp_;
        set fitstatistics;
        call symput("CRITLIKE",NegLogLike);
run;
%END;
%put &bmdl;
%mend;
```

E.2 Modifications to Wheeler's BMD Model SAS[®] code for the New Quantal Background Dose/Background Response models

This code combines the two macros in Wheeler(2005) and makes other modifications as needed. Wheeler provided code for all the BMD models described in his paper and the code below is modified from the appropriate model in Wheeler's code set.

```
/******* Versions of Wheeler's BMD code for the new Quantal Models ********/
/*
/*      Author:  Jinyg Feng of Battelle (Based on Wheeler's original Code)
/*
/***********************************************************************/
```

E.2.1 Multistage with Background Dose

```
/* Multistage with bg dose*/


%MACRO BMDMULTISTAGE(DATA,COUNT,TRIALS,DOSE,BMR,ADDEDRISK,DEGPOLY,CL);

data __one;
     set &data;
run;

/*SET UP THE POLYNOMIAL EFFECTS FOR THE MODEL*/

/*estimate the initial values for the parameter estimates*/
proc sort data = __one;
     by &dose;
run;

%LET PARMS = _GAMMA = &_GAMMA;
%LET BOUNDS = _GAMMA >= 0;
%DO I = 1 %TO &DEGPOLY;
     %IF ( &I = 1) %THEN %DO;
          %LET MODEL = _BDOSE1*(&DOSE+_GAMMA);
     %END; %ELSE %DO;
          %LET MODEL = &MODEL + _BDOSE&I*(&DOSE+_GAMMA)**&I;
     %END;
     %LET PARMS = &PARMS _BDOSE&I=0;
     %LET BOUNDS = &BOUNDS, _BDOSE&I >= 0;
%END;

ods listing ;
ods output fitstatistics = fitstatistics
              ParameterEstimates=ParameterEstimates;

/*FIT THE MODEL*/
proc nlmixed data= __one  CORR;
     bounds &bounds;
     _LINK = &MODEL;
     p = 1-EXP(-_LINK);
     MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
run;
```

```
/*PUT THE PARAMETER ESTIMATES INTO MACRO VARIABLES*/
data _temp_;
      set ParameterEstimates;
      if (_n_ = 1) then do;
                 call symput ("GAMMA",estimate);
      end;
      call symput(Parameter,estimate);
run;

/*ESTIMATE THE BMD AND THE LOWEST NON-NEGATIVE
  POLYNOMIAL IN THE MODEL*/
proc iml;
      use ParameterEstimates;
      read all var {"Estimate"} into estMat;
      close ParameterEstimates;
      gamma = estMat[1];
      powerTerms = estMat[2:nrow(estmat)];

    s1=0;
    do i = nrow(powerTerms) to 1 by -1;
                 s1=s1+(gamma**i)*powerTerms[i];
                 print s1;
      end;

      %IF (&ADDEDRISK = 1) %THEN %DO;
            z=log(exp(-s1)-&BMR);
      %END; %ELSE %DO;
            z=log(1-&BMR)-s1;
      %END;

/*INVERT THE VECTOR FROM HIGHEST TO LOWEST*/
      list = shape({0},nrow(powerTerms),1,0);
      do i = nrow(powerTerms) to 1 by -1;
            list[i] = powerTerms[nrow(powerTerms)-i + 1];
      end;
      list = list//z;
      print powerTerms;
      print list;

/*REMOVE ALL LEADING ZERO'S FROM THE MATRIX LIST*/
      fndPosCoef = 0;
      do while (fndPosCoef = 0);
            if (list[1,1] = 0) then list = list[2:nrow(list),1];

      end;

/*FIND THE ROOT OF THE POLYNOMIAL*/
      X = POLYROOT(list);
      print x;
      bmd = 0;
      do i = 1 to nrow(x);
          x[i,1]=x[i,1]-gamma;
            if (x[i,1]>0 & x[i,2] = 0) then bmd = x[i,1];
      end;

      create __predbmd from bmd;
```

```
      append from bmd;
      close  __predbmd;

/*FIND THE FIRST NON-NEGATIVE PARAMETER
USED AS A 'KEY OFF' FOR THE BMD ANALYSIS*/
      print powerTerms;
      found = 0;
      do i = 1 to nrow(powerTerms);
              if (powerTerms[i] > 0 & found =0) then do;
                      pos = i;
                      found=1;
              end;
      end;

      create __position from pos;
      append from pos;
      close  __position;
run;

/*SPECIFY THE DEGREE OF THE LOWEST POSTITIVE
  COEFFICIENT*/
data _temp;
      set __position;
      call symput("POLYDEG",COL1);
run;

/*SPECIFY THE NECESSARY VALUES FOR MACRO VARIABLES*/
data _temp_;
      set __predbmd;
      call symput("BMD",COL1);
      val = cinv(1-2*(1-&CL),1)*0.5;
      call symput("CRITVAL",val);
run;

data fitstatistics;
      set fitstatistics;
      format value best16.;
      informat value best16.;
      if (Descr = "-2 Log Likelihood");

      keep value NegLogLike;
run;

data _temp_;
              set fitstatistics;
              call symput("MLIKE",NegLogLike);
run;

/*SET UP THE NEW MODEL FOR THE MULTISTAGE
  MODEL*/

/* it is too difficult to solve for additional risk*/
/* for bmdl, it will only solve for extra risk */
/*%IF (&ADDEDRISK = 1) %THEN %DO; */
/* *  %LET A =(&BMR)/(1-_GAMMA);*/
/*    %LET z=log(exp(-s1)-&BMR); */
/*%END; %ELSE %DO; */
```

```
/*     %LET z=log(1-&BMR)-s1;   */
/*%END; */


%LET ERROR = 0;
%LET CRITLIKE = &MLIKE;


%LET I = 1;
%LET BMDL = &BMD;

%LET J = 1;
%LET _BOSE = _BDOSE;
/*PUT A LOWER BOUND ON THE BMD THIS IS DONE BY SLOWLY SHAVING
 2% OFF THE PRESENT ESTIMATE FOR THE BMD
 AFTER THIS IS FOUND WE FIND THE EXACT VALUE*/

%DO %WHILE (%SYSEVALF(&CRITLIKE - &MLIKE < &CRITVAL) AND &J < 600);

       %LET FIRSTTOKEN = 1;
       %LET LASTBMDL = &BMDL;
       %LET BMDL = %SYSEVALF(0.96*&BMDL);
       %LET BOUNDS = _GAMMA >=0;
       %LET PARMS = _GAMMA %str(=) &gamma;
       %let try1=0;
       %let try2=0;
       %LET SMODEL = ;
       /*SET UP THE INITIAL PARAMITER ESTIMATES AS WELL AS THE
         NUMERATOR TO THE PARAMETER SMODEL I.E.
                      BK = F(B1,B2..,BK-1,BK+1..,BN,GAMMA,BMD)/BMD^K
       */
       %DO I = 1 %TO &DEGPOLY;
            %IF (NOT(&I = &POLYDEG))%THEN %DO;

                      %LET SMODEL = -LOG(1-&BMR)-
_BDOSE&I*(&BMDL+_GAMMA)**&I+_BDOSE&I*_GAMMA**&I;
                      %LET FIRSTTOKEN = 2;
                 %END;
                 %ELSE %DO;
                      %LET SMODEL = &SMODEL  -
_BDOSE&I*(&BMDL+_GAMMA)**&I+_BDOSE&I*_GAMMA**&I;
                 %END;
                 %IF (%SYSEVALF(%left(&&_BDOSE&I) >= 0)) %THEN %DO;
                      %LET PARMS = &PARMS %left(_BDOSE&I%STR( =
)%left(&&_BDOSE&I));
                 %END; %ELSE %DO;
                      %LET PARMS = &PARMS %left(_BDOSE&I%STR( = )0);
                 %END;
                 %LET BOUNDS = &BOUNDS, _BDOSE&I >= 0;
            %END;
            %IF (&I = &POLYDEG) %THEN %DO;
                 %LET VARIABLE = _BDOSE&I;
            %END;
       %END;

       %DO I = 1 %TO &DEGPOLY;
           %let try1=&try1+_BDOSE&I*_GAMMA**&I;
```

```
                %let try2=&try2+_BDOSE&I*(&BMDL+_GAMMA)**&I;
        %end;
        %IF (&ADDEDRISK = 1) %THEN %DO;
          %let bounds= &bounds, log(exp(-&try1)-&bmr)+&try2=0;
        %end;

        ods listing close;
        ods output fitstatistics = fitstatistics
                        ParameterEstimates=ParameterEstimates;
        /*FIND THE PROFILE LIKELYHOOD FOR THE MODEL*/
        proc nlmixed data= __one;
                parms &parms;
                bounds &bounds;
                /*ENFORCE THE BOUND CONSTRAINTS ON THE VARIABLE*/
                &VARIABLE = (&SMODEL)/((&BMDL+_GAMMA)**&POLYDEG-
_GAMMA**&POLYDEG);
                call symput("_BDOSE%left(&POLYDEG)",&variable);
                call symput("variable",&variable);
                _LINK = &MODEL;
                P = 1-EXP(-_LINK);
                predict p out=pred;
                MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
        run;
        ods listing;
        data _temp_;
                set ParameterEstimates;
                if (_n_ = 1) then do;
                            call symput ("GAMMA",estimate);
                end;
                if (parameter = '_GAMMA') then do;
                      if (estimate = 0) then do; estimate = 0.01; end;
                end;
                call symput(Parameter,estimate);
        run;

        proc iml;
                use ParameterEstimates;
                read all var {"Estimate"} into estMat;
                close ParameterEstimates;
                gamma = estMat[1];


                /*find the lowest degree non zero parameter
                we use this one to 'Key off of' for our BMD analysis
                */
                temp = &variable;
                x = &polydeg;
                if (x = 1) then powerTerms = temp//powerTerms;
                else if (x > nrow(powerTerms)) then powerTerms =
powerTerms//temp;
                else powerTerms = powerTerms[1:(x-
1)]//temp//powerTerms[x:nrow(powerTerms)];
                found = 0;
                do i = 1 to nrow(powerTerms);
                            if (powerTerms[i] > 0 & found =0) then do;
                                  pos = i;
                                  found = 1;
```

```
                              end;
            end;
            create __position from pos;
            append from pos;
            close  __position;
        run;

        data _temp;
            set __position;
            call symput("POLYDEG",COL1);
        run;

        %let v =  &%left(_bdose%left(&degpoly));

        data fitstatistics;
            set fitstatistics;
            format value best16.;
            informat value best16.;
            if (Descr = "-2 Log Likelihood");

            keep value NegLogLike;
        run;

        data _temp_;
                set fitstatistics;
                call symput("CRITLIKE",NegLogLike);
        run;
        %put &critlike &mlike;
        %let J = %EVAL(&J + 1);

    %END;

%LET TOP = &LASTBMDL;
%LET BOTTOM = &BMDL;

DATA _TEMP_;
    X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
    X = X<0.0001;
    CALL SYMPUT("TEST",X);
RUN;

/*ZOOM IN ON THE BMDL USING A BINOMIAL HALVING SEARCH*/
%LET J = 1;
%DO %WHILE (&TEST = 0 AND &J < 600);

    %LET BMDL = %SYSEVALF((&TOP+&BOTTOM)/2);
    %LET FIRSTTOKEN = 1;
    %LET BOUNDS = _GAMMA >= 0;
    %LET PARMS = _GAMMA %str(=) &gamma;
    %LET SMODEL = ;
    %let try1=0;
    %let try2=0;
    %PUT &BMDL;

    /*SET UP THE INITIAL PARAMITER ESTIMATES AS WELL AS THE
      NUMERATOR TO THE PARAMETER SMODEL I.E.
                    BK = F(B1,B2..,BK-1,BK+1..,BN,GAMMA,BMD)/BMD^K
```

```sas
        */
        %DO I = 1 %TO &DEGPOLY;
                %IF (NOT(&I = &POLYDEG))%THEN %DO;

                            %LET SMODEL = -LOG(1-&BMR)-
_BDOSE&I*(&BMDL+_GAMMA)**&I+_BDOSE&I*_GAMMA**&I;
                            %LET FIRSTTOKEN = 2;
                    %END;
                    %ELSE %DO;
                            %LET SMODEL = &SMODEL  -
_BDOSE&I*(&BMDL+_GAMMA)**&I+_BDOSE&I*_GAMMA**&I;
                    %END;
                    %IF (%SYSEVALF(%left(&&_BDOSE&I) >= 0)) %THEN %DO;
                            %LET PARMS = &PARMS %left(_BDOSE&I%STR( =
)%left(&&_BDOSE&I));
                    %END; %ELSE %DO;
                            %LET PARMS = &PARMS %left(_BDOSE&I%STR( = )0);
                    %END;
                    %LET BOUNDS = &BOUNDS, _BDOSE&I >= 0;
                %END;
                %IF (&I = &POLYDEG) %THEN %DO;
                        %LET VARIABLE = _BDOSE&I;
                %END;
        %END;

            %DO I = 1 %TO &DEGPOLY;
            %let try1=&try1+_BDOSE&I*_GAMMA**&I;
                %let try2=&try2+_BDOSE&I*(&BMDL+_GAMMA)**&I;
        %end;
        %IF (&ADDEDRISK = 1) %THEN %DO;
          %let bounds= &bounds, log(exp(-&try1)-&bmr)+&try2=0;
        %end;

        ods listing close;
        ods output fitstatistics = fitstatistics
                        ParameterEstimates=ParameterEstimates;

        proc nlmixed data= __one;
                parms &parms;
                bounds &bounds;
                &VARIABLE = (&SMODEL)/((&BMDL+_GAMMA)**&POLYDEG-
_GAMMA**&POLYDEG);

                /*ENFORCE THE BOUND CONSTRAINTS ON THE VARIABLE*/
                call symput("_BDOSE%left(&POLYDEG)",&variable);
                call symput("variable",&variable);
                _LINK = &MODEL;
                p = 1-EXP(-_LINK);
                MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
        run;
        ods listing;

        data _temp_;
                set ParameterEstimates;
                if (_n_ = 1) then do;
                            call symput ("GAMMA",estimate);
                end;
```

```sas
            if (parameter = '_GAMMA') then do;
                  if (estimate = 0) then do; estimate = 0.01; end;
            end;
            call symput(Parameter,estimate);
      run;

      %let v =  &%left(_bdose%left(&degpoly));
      proc iml;
            use ParameterEstimates;
            read all var {"Estimate"} into estMat;
            close ParameterEstimates;
            gamma = estMat[1];
            powerTerms = estMat[2:nrow(estmat)];


            /*find the lowest degree non zero parameter
            we use this one to 'Key off of' for our BMD analysis
            */
            temp = &variable;

            x = &polydeg;
            if (x = 1) then powerTerms = temp//powerTerms;
            else if (x > nrow(powerTerms)) then powerTerms =
powerTerms//temp;
            else powerTerms = powerTerms[1:(x-
1)]//temp//powerTerms[x:nrow(powerTerms)];


            found = 0;
            do i = 1 to nrow(powerTerms);
                        if (powerTerms[i] > 0 & found =0) then do;
                              pos = i;
                              found = 1;
                        end;
            end;

            create __position from pos;
            append from pos;
            close  __position;
      run;

      data _temp;
            set __position;
            call symput("POLYDEG",COL1);
      run;
      data fitstatistics;
            set fitstatistics;
            format value best16.;
            informat value best16.;
            if (Descr = "-2 Log Likelihood");
            NegLogLike = value/2;
            keep value NegLogLike;
      run;

      data _temp_;
                set fitstatistics;
                call symput("CRITLIKE",NegLogLike);
```

```
        run;

        DATA _TEMP2_;
                X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);

                CALL SYMPUT("TEST",X2);
                Z = &CRITLIKE - &MLIKE-&CRITVAL;
                Z2 = Z < 0;
                CALL SYMPUT("LOCALTEST",Z2);
                output;
        RUN;

        %IF (&LOCALTEST = 1) %THEN %DO;
                %LET TOP = &BMDL;
        %END;
        %IF (&LOCALTEST =0) %THEN %DO;
                %LET BOTTOM = &BMDL;
        %END;
        %LET J = %EVAL(&J+1);
%END;

data _temp_;
        set &data;
        retain value 0;
        value = value + &trials;
        call symput("NVALUE",value);
run;

data BMDMULTI;
        NAME = "MULTISTAGE";
        BMD = &BMD;
        BMDL = &BMDL;
        MAXLIKE= 2*&MLIKE;
        N = &NVALUE;
        BIC = MAXLIKE + (1+&DEGPOLY)*LOG(N);
RUN;

%MEND;

data EGdata;
input dose obs n;
cards;
        0 2 10
        0.5 2 10
        1.0 2 10
        2.0 6 10
        4.0 9 10
;
run;

%BMDMULTISTAGE(EGdata,obs,n,DOSE,0.1,0,2,0.95);
```

E.2.1 Log-Logistic with Background Dose

```
/* Log-Logistic with bg dose*/

%MACRO BMDLOGLOGISTIC(DATA,COUNT,TRIALS,DOSE,BMR,ADDEDRISK,CL);

data __one;
      set &data;
run;

%LET PARMS = _ALPHA = -5 TO 0 BY 0.5 _BDOSE = 1 TO 5 BY 0.5 _GAMMA = 0.01 to
1 by 0.05;
%LET BOUNDS = _GAMMA >0, _GAMMA<=1;
%LET MODEL = _ALPHA + _BDOSE*(LOG(&DOSE+_GAMMA));

ods output fitstatistics = fitstatistics
                ParameterEstimates=ParameterEstimates;

/*FIT THE MODEL*/
proc nlmixed data= __one corr maxiter=100000;
      parms &parms;
      bounds &bounds;


      _LINK = &MODEL;
      P = 1/(1+EXP(-_LINK));

      %IF (&ADDEDRISK = 1) %THEN %DO;
            _X = exp(-(_ALPHA+_BDOSE*LOG(_GAMMA)));
            _y = log((&BMR+&BMR*_x+1)/(_x-&BMR-&BMR*_x))-_ALPHA;
            _BMD = _y/_BDOSE;
            _BMD = exp(_BMD)-_GAMMA;
      %END; %ELSE %DO;
            _X = exp(-(_ALPHA+_BDOSE*LOG(_GAMMA)));
            _y=exp(-_ALPHA)/((1+_x)/((&BMR*_x)+1)-1);
            _BMD = LOG(_y)/_BDOSE;
            _BMD = exp(_BMD)-_GAMMA;
      %END;
      predict p out=pred;
      CALL SYMPUT("BMD",_BMD);
      MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
run;

data BMDLOGLOGIT;
      NAME = "LOGLOGISTIC";
      BMD = &BMD;
*     BMDL = &BMDL;
*     MAXLIKE= 2*&MLIKE;
*     N = &NVALUE;
*     BIC = MAXLIKE + 3*LOG(N);
RUN;

%MEND;

data EGdata;
input dose obs n;
```

```
cards;
0 2 10
0.5  2 10
1.0  2 10
2.0  6 10
4.0  9 10
;
run;

%BMDLOGLOGISTIC(Egdata,obs,n,DOSE,0.1,0,0.95);
```

E.2.3 Gamma with Background Dose

```
/* Gamma with bg dose*/

%MACRO BMDGAMMA_BGD(DATA,COUNT,TRIALS,DOSE,ALPHA_LIM,BMR,ADDEDRISK,CL);

/*SET UP THE POLYNOMIAL EFFECTS FOR THE MODEL*/
%LET PARMS = _ALPHA = 0.5 TO 20 BY 0.5
             _BETA = 0.1 TO 5.0 BY 0.1
             _NU = 0.1 TO 20.0 BY 0.5;

ods output FitStatistics = FitStatistics
           ParameterEstimates = ParameterEstimates
           CorrMatParmEst = CORGAMMA_BGD;

/*FIT THE MODEL*/
proc nlmixed data=&DATA maxfunc=100000 maxiter=10000 HESS CORR;
    parms &PARMS;

    bounds _ALPHA >= &ALPHA_LIM, _BETA > 0, _NU >= 0;
    P = PROBGAM(_BETA * (&DOSE + _NU),_ALPHA);

    %IF (&ADDEDRISK = 1) %THEN %DO;
        _X = GAMINV((&BMR + PROBGAM(_BETA * _NU,_ALPHA)),_ALPHA);
    %END;
    %ELSE %DO;
        _X = GAMINV((&BMR + (1-&BMR)*PROBGAM(_BETA * _NU,_ALPHA)),_ALPHA);
    %END;
    _BMD = _X/_BETA-_NU;

    predict p out=pred;
    CALL SYMPUT("BMD",_BMD);
    MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
run;

ods output close;

data ESTGAMMA_BGD;
    set ParameterEstimates;
run;

data FitStatistics;
    set FitStatistics;
    format value best16.;
    informat value best16.;
    if (Descr = "-2 Log Likelihood");
    NegLogLike = value/2;
    keep value NegLogLike;
run;

data _temp_;
    set FitStatistics;
    call symput("MLIKE",NegLogLike);

    call symput("CRITVAL",val);
```

```
run;

%LET ERROR = 0;
%LET CRITLIKE = &MLIKE;
%LET I = 1;
%LET BMDL = &BMD;
%LET J = 1;
%LET _BOSE = _BDOSE;

/*PUT A LOWER BOUND ON THE BMD THIS IS DONE BY SLOWLY SHAVING
 2% OFF THE PRESENT ESTIMATE FOR THE BMD
 AFTER THIS IS FOUND WE FIND THE EXACT VALUE*/

%DO %WHILE (%SYSEVALF(&CRITLIKE - &MLIKE < &CRITVAL) AND &J < 200);

    %LET FIRSTTOKEN = 1;
    %LET LASTBMDL = &BMDL;
    %LET BMDL = %SYSEVALF(1.02*&BMDL);
    %LET SMODEL = ;

    data pe;
        set ParameterEstimates;
        %IF (&J = 1) %THEN %DO;
            retain alpha beta;
            drop alpha beta;
            if (Parameter = '_ALPHA') then alpha = Estimate;
            else if (Parameter='_BETA') then do;
                beta = Estimate;
                delete;
            end;
            else if (Parameter = '_NU') then do;
                Parameter = '_LAMBDA';
                Estimate = beta * Estimate;
            end;
        %END;
        if (Parameter = '_ALPHA' & Estimate = &ALPHA_LIM) then
            Estimate = &ALPHA_LIM + 0.001;
        else if (Parameter = '_LAMBDA'  & Estimate = 0) then
            Estimate = 0.001;
    run;

    ods listing close;
    ods output FitStatistics = FitStatistics
               ParameterEstimates = ParameterEstimates;

    proc nlmixed data=&DATA;
        parms / data = pe;

        bounds _ALPHA >= &ALPHA_LIM, _LAMBDA >= 0;
        %IF (&ADDEDRISK = 1) %THEN %DO;
            _X = GAMINV((&BMR + PROBGAM(_LAMBDA,_ALPHA)),_ALPHA);
        %END;
        %ELSE %DO;
            _X = GAMINV((&BMR + (1-&BMR)*PROBGAM(_LAMBDA,_ALPHA)),_ALPHA);
        %END;
        _BETA = (_X - _LAMBDA) / &BMDL;
        P = PROBGAM(_BETA * &DOSE + _LAMBDA,_ALPHA);
```

```sas
        MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
    run;
    ods listing;

    data FitStatistics;
        set FitStatistics;
        format value best16.;
        informat value best16.;
        if (Descr = "-2 Log Likelihood");
        NegLogLike = value/2;
        keep value NegLogLike;
    run;

    data _temp_;
        set FitStatistics;
        call symput("CRITLIKE",NegLogLike);
    run;
    %let J = %EVAL(&J + 1);

%END;

%LET TOP = &LASTBMDL;
%LET BOTTOM = &BMDL;

DATA _TEMP_;
    X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
    X = X<0.0001;
    CALL SYMPUT("TEST",X);
RUN;

/*ZOOM IN ON THE BMDL USING A BINOMIAL HALVING SEARCH*/
%LET J = 1;
%DO %WHILE (&TEST = 0 AND &J < 20);

    %LET BMDL = %SYSEVALF((&TOP+&BOTTOM)/2);
    %LET FIRSTTOKEN = 1;
    %LET SMODEL = ;
    %PUT &BMDL;

    data pe;
        set ParameterEstimates;
        if (Parameter = '_ALPHA' & Estimate = &ALPHA_LIM) then
            Estimate = &ALPHA_LIM + 0.001;
        else if (Parameter = '_LAMBDA'  & Estimate = 0) then

    run;

    ods listing close;
    ods output FitStatistics = FitStatistics
                ParameterEstimates = ParameterEstimates;

    proc nlmixed data=&DATA;
        parms / data=pe;

        bounds _ALPHA >= &ALPHA_LIM, _LAMBDA >= 0;
        %IF (&ADDEDRISK = 1) %THEN %DO;
```

```
            _X = GAMINV((&BMR + PROBGAM(_LAMBDA,_ALPHA)),_ALPHA);
        %END;
        %ELSE %DO;
            _X = GAMINV((&BMR + (1-&BMR)*PROBGAM(_LAMBDA,_ALPHA)),_ALPHA);
        %END;
        _BETA = (_X - _LAMBDA) / &BMDL;
        P = PROBGAM(_BETA * &DOSE + _LAMBDA,_ALPHA);

        MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
    run;
    ods listing;

    data FitStatistics;
        set FitStatistics;
        format value best16.;
        informat value best16.;
        if (Descr = "-2 Log Likelihood");

        keep value NegLogLike;
    run;

    data _temp_;
        set FitStatistics;
        call symput("CRITLIKE",NegLogLike);
    run;

    DATA _TEMP2_;
        X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
        X2 = X < 0.0001;
        CALL SYMPUT("TEST",X2);
        Z = &CRITLIKE - &MLIKE-&CRITVAL;
        Z2 = Z < 0;
        CALL SYMPUT("LOCALTEST",Z2);
        output;
    RUN;

    %IF (&LOCALTEST = 1) %THEN %DO;
        %LET TOP = &BMDL;
    %END;
    %IF (&LOCALTEST =0) %THEN %DO;
        %LET BOTTOM = &BMDL;
    %END;
    %LET J = %EVAL(&J+1);
%END;

data _temp_;
    set &DATA;
    retain value 0;
    value = value + &trials;
    call symput("NVALUE",value);
run;

data BMDGAMMA_BGD;
    NAME = "BACKGROUND DOSE GAMMA";
    BMD = &BMD;
    BMDL = &BMDL;
    MAXLIKE= 2*&MLIKE;
```

```
    N = &NVALUE;
    AIC = MAXLIKE + 2*3;
    BIC = MAXLIKE + 3*log(N);
RUN;

proc datasets;
    delete FitStatistics ParameterEstimates Pred _temp_ pe _temp2_;
run;
quit;

%MEND BMDGAMMA_BGD;




data testdata;
    input dose obs;
    alpha = 2.5; beta = 0.3; nu = 1.5;
    n = round(obs * PROBGAM(beta * (dose + nu),alpha));
    keep dose n obs;
cards;
     0   100
     0.5 100
     1.0 100
     2.0 100
     4.0 100
;
run;


%BMDGAMMA_BGD(testdata,n,obs,dose,0,0.1,0,0.95);
```

E.2.4 Weibull with Background Dose

```
/* Weibull with bg dose*/


%MACRO BMDWEIBULL_BGD(DATA,COUNT,TRIALS,DOSE,ALPHA_LIM,BMR,ADDEDRISK,CL);

/*SET UP THE POLYNOMIAL EFFECTS FOR THE MODEL*/
%LET PARMS = _ALPHA = 0.5 TO 10 BY 0.5
             _BETA = 0.1 TO 2.0 BY 0.1
             _NU = 0.1 TO 2.0 BY 0.1;


ods output FitStatistics = FitStatistics
           ParameterEstimates = ParameterEstimates
           CorrMatParmEst = CORWEIBULL_BGD;

/*FIT THE MODEL*/
proc nlmixed data=&DATA maxfunc=100000 maxiter=10000 HESS CORR;
    parms &PARMS;

    bounds _ALPHA >= &ALPHA_LIM, _BETA > 0, _NU >= 0;
    P = 1 - EXP(-_BETA * (&DOSE + _NU)**_ALPHA);

    %IF (&ADDEDRISK = 1) %THEN %DO;
        _X = -LOG(EXP(-_BETA * _NU**_ALPHA)-&BMR)/_BETA;
    %END;
    %ELSE %DO;
        _X = -LOG(1-&BMR)/_BETA + _NU**_ALPHA;
    %END;
    _BMD = _X**(1/_ALPHA)-_NU;

    predict p out=pred;
    CALL SYMPUT("BMD",_BMD);
    MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
run;

ods output close;

data ESTWEIBULL_BGD;
    set ParameterEstimates;
run;

data FitStatistics;
    set FitStatistics;
    format value best16.;
    informat value best16.;
    if (Descr = "-2 Log Likelihood");
    NegLogLike = value/2;
    keep value NegLogLike;
run;

data _temp_;
    set FitStatistics;
    call symput("MLIKE",NegLogLike);
```

```
      call symput("CRITVAL",val);
run;

%LET ERROR = 0;
%LET CRITLIKE = &MLIKE;
%LET I = 1;
%LET BMDL = &BMD;
%LET J = 1;
%LET _BOSE = _BDOSE;

/*PUT A LOWER BOUND ON THE BMD THIS IS DONE BY SLOWLY SHAVING
 2% OFF THE PRESENT ESTIMATE FOR THE BMD
 AFTER THIS IS FOUND WE FIND THE EXACT VALUE*/

%DO %WHILE (%SYSEVALF(&CRITLIKE - &MLIKE < &CRITVAL) AND &J < 200);

    %LET FIRSTTOKEN = 1;
    %LET LASTBMDL = &BMDL;
    %LET BMDL = %SYSEVALF(0.98*&BMDL);
    %LET SMODEL = ;

    data pe;
        set ParameterEstimates;
        %IF (&ADDEDRISK = 1) %THEN %DO;
            %IF (&J = 1) %THEN %DO;
                retain alpha beta;
                drop alpha beta;
                if (Parameter = '_ALPHA') then alpha = Estimate;
                else if (Parameter='_BETA') then do;
                    beta = Estimate;
                    delete;
                end;
                else if (Parameter = '_NU') then do;
                    Parameter = '_KAPPA';
                    Estimate = beta * Estimate**alpha;
                end;
            %END;
            if (Parameter = '_ALPHA' & Estimate = &ALPHA_LIM) then
                Estimate = &ALPHA_LIM + 0.001;
            else if (Parameter = '_KAPPA'  & Estimate = 0) then
                Estimate = 0.001;
        %END;
        %ELSE %DO;
            if (Parameter = '_ALPHA' & Estimate = &ALPHA_LIM) then
                Estimate = &ALPHA_LIM + 0.001;
            else if (Parameter='_BETA') then delete;

                Estimate = 0.001;
        %END;
    run;

    ods listing close;
    ods output FitStatistics = FitStatistics
               ParameterEstimates = ParameterEstimates;

    proc nlmixed data=&DATA;
        parms / data = pe;
```

```
        %IF (&ADDEDRISK = 1) %THEN %DO;
            bounds _ALPHA >= &ALPHA_LIM, _KAPPA >= 0;
            _BETA = (((-LOG(EXP(-_KAPPA) - &BMR))**(1/_ALPHA) -
_KAPPA**(1/_ALPHA)) / &BMDL)**_ALPHA;
            P = 1 - EXP(-(&DOSE * _BETA**(1/_ALPHA) +
_KAPPA**(1/_ALPHA))**_ALPHA);
        %END;
        %ELSE %DO;
            bounds _ALPHA >= &ALPHA_LIM, _NU >= 0;
            _BETA = -LOG(1-&BMR)/((&BMDL+_NU)**_ALPHA - _NU**_ALPHA);

        %END;
        MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
    run;
    ods listing;

    data FitStatistics;
        set FitStatistics;
        format value best16.;
        informat value best16.;
        if (Descr = "-2 Log Likelihood");
        NegLogLike = value/2;
        keep value NegLogLike;
    run;

    data _temp_;
        set FitStatistics;
        call symput("CRITLIKE",NegLogLike);
    run;
    %let J = %EVAL(&J + 1);

%END;

%LET TOP = &LASTBMDL;
%LET BOTTOM = &BMDL;

DATA _TEMP_;
    X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
    X = X<0.0001;
    CALL SYMPUT("TEST",X);
RUN;

/*ZOOM IN ON THE BMDL USING A BINOMIAL HALVING SEARCH*/
%LET J = 1;
%DO %WHILE (&TEST = 0 AND &J < 20);

    %LET BMDL = %SYSEVALF((&TOP+&BOTTOM)/2);
    %LET FIRSTTOKEN = 1;
    %LET SMODEL = ;
    %PUT &BMDL;

    data pe;
        set ParameterEstimates;
        if (Parameter = '_ALPHA' & Estimate = &ALPHA_LIM) then
            Estimate = &ALPHA_LIM + 0.001;
        %IF (&ADDEDRISK = 1) %THEN %DO;
```

```sas
        else if (Parameter = '_KAPPA'  & Estimate = 0) then
            Estimate = 0.001;
    %END;
    %ELSE %DO;
        else if (Parameter = '_NU' & Estimate = 0) then
            Estimate = 0.001;
    %END;
run;

ods listing close;
ods output FitStatistics = FitStatistics
        ParameterEstimates = ParameterEstimates;

proc nlmixed data=&DATA;
    parms / data=pe;

    %IF (&ADDEDRISK = 1) %THEN %DO;
        bounds _ALPHA >= &ALPHA_LIM, _KAPPA >= 0;
        _BETA = (((-LOG(EXP(-_KAPPA) - &BMR))**(1/_ALPHA) -
_KAPPA**(1/_ALPHA)) / &BMDL)**_ALPHA;
        P = 1 - EXP(-(&DOSE * _BETA**(1/_ALPHA) +
_KAPPA**(1/_ALPHA))**_ALPHA);
    %END;
    %ELSE %DO;
        bounds _ALPHA >= &ALPHA_LIM, _NU >= 0;
        _BETA = -LOG(1-&BMR)/((&BMDL+_NU)**_ALPHA - _NU**_ALPHA);
        P = 1 - EXP(-_BETA * (&DOSE + _NU)**_ALPHA);
    %END;

    MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
run;
ods listing;

data FitStatistics;
    set FitStatistics;
    format value best16.;
    informat value best16.;
    if (Descr = "-2 Log Likelihood");

    keep value NegLogLike;
run;

data _temp_;
    set FitStatistics;
    call symput("CRITLIKE",NegLogLike);
run;

DATA _TEMP2_;
    X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
    X2 = X < 0.0001;
    CALL SYMPUT("TEST",X2);
    Z = &CRITLIKE - &MLIKE-&CRITVAL;
    Z2 = Z < 0;
    CALL SYMPUT("LOCALTEST",Z2);
    output;
RUN;
```

```
    %IF (&LOCALTEST = 1) %THEN %DO;
         %LET TOP = &BMDL;
    %END;
    %IF (&LOCALTEST =0) %THEN %DO;
         %LET BOTTOM = &BMDL;
    %END;
    %LET J = %EVAL(&J+1);
%END;

data _temp_;
    set &DATA;
    retain value 0;
    value = value + &trials;
    call symput("NVALUE",value);
run;

data BMDWEIBULL_BGD;
    NAME = "BACKGROUND DOSE WEIBULL";
    BMD = &BMD;
    BMDL = &BMDL;
    MAXLIKE= 2*&MLIKE;
    N = &NVALUE;
    AIC = MAXLIKE + 2*3;
    BIC = MAXLIKE + 3*log(N);
RUN;

proc datasets;
    delete FitStatistics ParameterEstimates Pred _temp_ pe _temp2_;
run;
quit;

%MEND BMDWEIBULL_BGD;
```

E.2.5 Logistic with Background Response

```sas
/* Logistic with bg response */

%MACRO BMDLOGIT(DATA,COUNT,TRIALS,DOSE,BMR,ADDEDRISK,CL);

data __one;
      set &data;
run;

%LET PARMS = _ALPHA = -20 TO 0 BY 1 _BDOSE = 1 TO 5 BY 0.5 _GAMMA =0 to 1 by
0.1;
%LET BOUNDS =  _BDOSE <= 40, _GAMMA<=1, _GAMMA>=0;
%LET MODEL = _ALPHA + _BDOSE*(&DOSE);

ods output fitstatistics = fitstatistics
                ParameterEstimates=ParameterEstimates;

/*FIT THE MODEL*/
proc nlmixed data= __one corr;
      parms &parms;
      bounds &bounds;
      _LINK = &MODEL;
      P = _GAMMA+(1-_GAMMA)/(1+exp(-_LINK));
      %IF (&ADDEDRISK = 1) %THEN %DO;
            _Z = (1-_GAMMA)*(1+exp(-_ALPHA))/((1-_GAMMA)+&BMR*(1+EXP(-
_ALPHA)));
            _X = log((_Z-1)/exp(-_ALPHA));
            _BMD = -_X/_BDOSE;
      %END; %ELSE %DO;
            _Z = &BMR;
            _X = LOG((1-_Z)/(1+_Z*EXP(-_ALPHA)));
            _BMD = -_X/_BDOSE;
      %END;
      predict p out=pred;
      CALL SYMPUT("BMD",_BMD);
      MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
run;

ods listing close;
data fitstatistics;
      set fitstatistics;
      format value best16.;
      informat value best16.;
      if (Descr = "-2 Log Likelihood");
      NegLogLike = value/2;
      keep value NegLogLike;
run;
data _temp_;
            set fitstatistics;
            call symput("MLIKE",NegLogLike);
            val = cinv(1-2*(1-&CL),1)*0.5;
            call symput("CRITVAL",val);
run;


%LET ERROR = 0;
```

```
%LET CRITLIKE = &MLIKE;
%LET I = 1;
%LET BMDL = &BMD;
%LET J = 1;
%LET _BOSE = _BDOSE;
/*PUT A LOWER BOUND ON THE BMD THIS IS DONE BY SLOWLY SHAVING
 2% OFF THE PRESENT ESTIMATE FOR THE BMD
 AFTER THIS IS FOUND WE FIND THE EXACT VALUE*/

%DO %WHILE (%SYSEVALF(&CRITLIKE - &MLIKE < &CRITVAL) AND &J < 200);

     %LET FIRSTTOKEN = 1;
     %LET LASTBMDL = &BMDL;
     %LET BMDL = %SYSEVALF(1.02*&BMDL);
     %LET SMODEL = ;

     data pe; set ParameterEstimates; if(Parameter='_BDOSE')then delete;
run;

     ods listing close;
ods output fitstatistics = fitstatistics
             ParameterEstimates=ParameterEstimates;

     proc nlmixed data= __one;
          parms /data=pe;
       bounds _GAMMA>=0,_GAMMA<=1;

          %IF (&ADDEDRISK = 1) %THEN %DO;
               Z = (1-_GAMMA)*(1+exp(-_ALPHA))/((1-_GAMMA)+&BMR*(1+EXP(-
_ALPHA)));
               _X = log((_Z-1)/exp(-_ALPHA));
               _BDOSE = -_X/&BMDL;
          %END; %ELSE %DO;
               _Z = &BMR;
               _X = LOG((1-_Z)/(1+_Z*EXP(-_ALPHA)));
               _BDOSE = -_X/&BMDL;
          %END;
          _LINK = &MODEL;
          P = _GAMMA+(1-_GAMMA)/(1+exp(-_LINK));
          MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
     run;

     ods listing;
     data fitstatistics;
          set fitstatistics;
          format value best16.;
          informat value best16.;
          if (Descr = "-2 Log Likelihood");

          keep value NegLogLike;
     run;

     data _temp_;
               set fitstatistics;
               call symput("CRITLIKE",NegLogLike);
     run;
     %let J = %EVAL(&J + 1);
```

```
%END;

%LET TOP = &LASTBMDL;
%LET BOTTOM = &BMDL;

DATA _TEMP_;
      X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
      X = X<0.0001;
      CALL SYMPUT("TEST",X);
RUN;

/*ZOOM IN ON THE BMDL USING A BINOMIAL HALVING SEARCH*/
%LET J = 1;
%DO %WHILE (&TEST = 0 AND &J < 20);

      %LET BMDL = %SYSEVALF((&TOP+&BOTTOM)/2);
      %LET FIRSTTOKEN = 1;
      %LET BOUNDS = _GAMMA >= 0, _GAMMA <= 1;
      %LET SMODEL = ;
      %PUT &BMDL;

      data pe; set ParameterEstimates; if(Parameter='_BDOSE')then delete;
         if (Parameter='_GAMMA') and estimate<0 then estimate=0; run;

      ods listing close;
ods output fitstatistics = fitstatistics
             ParameterEstimates=ParameterEstimates;

      proc nlmixed data= __one;
            parms /data=pe;
            bounds _GAMMA>=0,_GAMMA<=1;

            %IF (&ADDEDRISK = 1) %THEN %DO;
                  Z = (1-_GAMMA)*(1+exp(-_ALPHA))/((1-_GAMMA)+&BMR*(1+EXP(-
_ALPHA)));
                  _X = log((_Z-1)/exp(-_ALPHA));
                  _BDOSE = -_X/&BMDL;
            %END; %ELSE %DO;
                  _Z = &BMR;
                  _X = LOG((1-_Z)/(1+_Z*EXP(-_ALPHA)));
                  _BDOSE = -_X/&BMDL;
            %END;
            _LINK = &MODEL;
            P = _GAMMA+(1-_GAMMA)/(1+exp(-_LINK));
            MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
      run;

      ods listing;

      data fitstatistics;
            set fitstatistics;
            format value best16.;
            informat value best16.;
            if (Descr = "-2 Log Likelihood");

            keep value NegLogLike;
```

```
        run;

        data _temp_;
                set fitstatistics;
                call symput("CRITLIKE",NegLogLike);
        run;

        DATA _TEMP2_;
                X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
                X2 = X < 0.0001;
                CALL SYMPUT("TEST",X2);
                Z = &CRITLIKE - &MLIKE-&CRITVAL;
                Z2 = Z < 0;
                CALL SYMPUT("LOCALTEST",Z2);
                output;
        RUN;

        %IF (&LOCALTEST = 1) %THEN %DO;
                %LET TOP = &BMDL;
        %END;
        %IF (&LOCALTEST =0) %THEN %DO;
                %LET BOTTOM = &BMDL;
        %END;
        %LET J = %EVAL(&J+1);
%END;

data _temp_;
        set &data;
        retain value 0;
        value = value + &trials;
        call symput("NVALUE",value);
run;

data BMDLOGIT;
        NAME = "LOGISTIC";
        BMD = &BMD;
        BMDL = &BMDL;
        MAXLIKE= 2*&MLIKE;
        N = &NVALUE;
        BIC = MAXLIKE + 2*LOG(N);
RUN;
%MEND;

data EGdata;
input dose obs n;
cards;
0 2 10
0.5 2 10
1.0 2 10
2.0 6 10
4.0 9 10
;
run;

%BMDLOGIT(Egdata,obs,n,DOSE,0.1,0,0.95);
```

E.2.6 Probit with Background Response

```
/* Probit with bg response*/
%MACRO BMDPROBIT(DATA,COUNT,TRIALS,DOSE,BMR,ADDEDRISK,CL);

data __one;
      set &data;
run;

%LET PARMS = _GAMMA =0 To 1 by 0.1 _ALPHA = -10 TO 0 BY 0.1 _BDOSE = 1 TO 10
BY 0.1;
%LET BOUNDS =  _BDOSE <= 40, _GAMMA>=0, _GAMMA<=1;
%LET MODEL = _ALPHA + _BDOSE*(&DOSE);


ods output fitstatistics = fitstatistics
               ParameterEstimates=ParameterEstimates;
/*FIT THE MODEL*/
proc nlmixed data= __one corr;
      parms &parms;
      bounds &bounds;
      _LINK = &MODEL;
      P = _GAMMA + (1-_GAMMA)*PROBNORM(_LINK);
      %IF (&ADDEDRISK = 1) %THEN %DO;
            _X = PROBIT(&BMR/(1-_GAMMA)+PROBNORM(_ALPHA))-_ALPHA;
            _BMD = _X/_BDOSE;
      %END; %ELSE %DO;
            _X = PROBIT(&BMR*(1-PROBNORM(_ALPHA))+PROBNORM(_ALPHA))-_ALPHA;
            _BMD = _X/_BDOSE;
      %END;
      predict p out=pred;
      CALL SYMPUT("BMD",_BMD);
      MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
run;
data fitstatistics;
      set fitstatistics;
      format value best16.;
      informat value best16.;
      if (Descr = "-2 Log Likelihood");
      NegLogLike = value/2;
      keep value NegLogLike;
run;
data _temp_;
            set fitstatistics;
            call symput("MLIKE",NegLogLike);
            val = cinv(1-2*(1-&CL),1)*0.5;
            call symput("CRITVAL",val);
run;

ods listing close;

%LET ERROR = 0;
%LET CRITLIKE = &MLIKE;
%LET I = 1;
%LET BMDL = &BMD;
%LET J = 1;
```

```sas
%LET _BOSE = _BDOSE;
/*PUT A LOWER BOUND ON THE BMD THIS IS DONE BY SLOWLY SHAVING
 2% OFF THE PRESENT ESTIMATE FOR THE BMD
 AFTER THIS IS FOUND WE FIND THE EXACT VALUE*/

%DO %WHILE (%SYSEVALF(&CRITLIKE - &MLIKE < &CRITVAL) AND &J < 200);

      %LET FIRSTTOKEN = 1;
      %LET LASTBMDL = &BMDL;
      %LET BMDL = %SYSEVALF(1.02*&BMDL);
      %LET SMODEL = ;

      data pe; set ParameterEstimates; if(Parameter='_BDOSE')then delete;
run;

      ods listing close;
ods output fitstatistics = fitstatistics
              ParameterEstimates=ParameterEstimates;

      proc nlmixed data= __one;
            parms /data=pe; ;
            %IF (&ADDEDRISK = 1) %THEN %DO;
                  _X = PROBIT(&BMR/(1-_GAMMA)+PROBNORM(_ALPHA))-_ALPHA;
                  _BDOSE = _X/&BMDL;
            %END; %ELSE %DO;
                  _X = PROBIT(&BMR*(1-PROBNORM(_ALPHA))+PROBNORM(_ALPHA))-
_ALPHA;
                  _BDOSE = _X/&BMDL;
            %END;
            _LINK = &MODEL;
            P = _GAMMA + (1-_GAMMA)*PROBNORM(_LINK);
            MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
      run;

      ods listing;
      data fitstatistics;
            set fitstatistics;
            format value best16.;
            informat value best16.;
            if (Descr = "-2 Log Likelihood");

            keep value NegLogLike;
      run;

      data _temp_;
                set fitstatistics;
                call symput("CRITLIKE",NegLogLike);
      run;
      %let J = %EVAL(&J + 1);

%END;

%LET TOP = &LASTBMDL;
%LET BOTTOM = &BMDL;

DATA _TEMP_;
      X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
```

```
        X = X<0.0001;
        CALL SYMPUT("TEST",X);
RUN;

/*ZOOM IN ON THE BMDL USING A BINOMIAL HALVING SEARCH*/
%LET J = 1;
%DO %WHILE (&TEST = 0 AND &J < 20);

        %LET BMDL = %SYSEVALF((&TOP+&BOTTOM)/2);
        %LET FIRSTTOKEN = 1;
        %LET BOUNDS = _GAMMA >= 0, _GAMMA <= 1;
        %LET SMODEL = ;
        %PUT &BMDL;

        data pe; set ParameterEstimates; if(Parameter='_BDOSE')then delete;
run;

        ods listing close;
ods output fitstatistics = fitstatistics
                ParameterEstimates=ParameterEstimates;

        proc nlmixed data= __one;
                parms /data=pe; ;
                %IF (&ADDEDRISK = 1) %THEN %DO;
                        _X = PROBIT(&BMR/(1-_GAMMA)+PROBNORM(_ALPHA))-_ALPHA;
                        _BDOSE = _X/&BMDL;
                %END; %ELSE %DO;
                        _X = PROBIT(&BMR*(1-PROBNORM(_ALPHA))+PROBNORM(_ALPHA))-
_ALPHA;
                        _BDOSE = _X/&BMDL;
                %END;
                _LINK = &MODEL;
                P = _GAMMA + (1-_GAMMA)*PROBNORM(_LINK);
                MODEL &COUNT ~ BINOMIAL(&TRIALS,P);
        run;
        ods listing;

        data fitstatistics;
                set fitstatistics;
                format value best16.;
                informat value best16.;
                if (Descr = "-2 Log Likelihood");

                keep value NegLogLike;
        run;

        data _temp_;
                        set fitstatistics;
                        call symput("CRITLIKE",NegLogLike);
        run;

        DATA _TEMP2_;
                X = ABS(&CRITLIKE - &MLIKE-&CRITVAL);
                X2 = X < 0.0001;
                CALL SYMPUT("TEST",X2);
                Z = &CRITLIKE - &MLIKE-&CRITVAL;
                Z2 = Z < 0;
```

```
                CALL SYMPUT("LOCALTEST",Z2);
                output;
        RUN;

        %IF (&LOCALTEST = 1) %THEN %DO;
                %LET TOP = &BMDL;
        %END;
        %IF (&LOCALTEST =0) %THEN %DO;
                %LET BOTTOM = &BMDL;
        %END;
        %LET J = %EVAL(&J+1);
%END;
data _temp_;
        set &data;
        retain value 0;
        value = value + &trials;
        call symput("NVALUE",value);
run;

data BMDPROBIT;
        NAME = "PROBIT";
        BMD = &BMD;
        BMDL = &BMDL;
        MAXLIKE= 2*&MLIKE;
        N = &NVALUE;
        BIC = MAXLIKE + 2*LOG(N);
RUN;

%MEND;

data EGdata;
input dose obs n;
cards;
0 2 10
0.5 2 10
1.0 2 10
2.0 6 10
4.0 9 10
;
run;

%BMDPROBIT(EGdata,obs,n,DOSE,0.1,1,0.95);
```